```
DEFINE info_tag
  PARAMETER uniqID
  PARAMETER sub_filename
  PARAMETER rev_type

{
all revisions are indicated in rev block of sheet one
all other sheets have all revisions and say "see sheet one" for the PCO#
-when dealing with the new IPG standard border you will need to add 10 to
   the x and to the y dimensions on all LL and UR coordinates of rev and
   title block fields. also the UR of the border will be +20, +20 from LL.
-also the upper right coordinate
}

  LOCAL Input_file
  LOCAL message_file
  LOCAL xys
  LOCAL bd_size
  LOCAL currentInfos
  LOCAL infoLine
  LOCAL blockTimer
  LOCAL mynotes
  LOCAL mysheets
  LOCAL blanks
  LOCAL border_width
  LOCAL myLine

  LET mmd "OFF"
  LET pnts "OFF"
  LET orgFile ('/tmp/'+sub_filename+':s.inf')
  LET newFile (TMPMDX+'/'+sub_filename+'.inf')
  LET blockTimer ('/tmp/blockTimer.$LOGNAME')
  LET currentInfos ('/tmp/currentInfos.'+uniqID)
  LET Input_file ('/tmp/Me30_parts_list.'+uniqID)
  LET message_file ('/tmp/message.'+uniqID)
  LET mynotes ('/tmp/mynotes.'+uniqID)
  LET xys ('/tmp/xys.'+uniqID)
  LET cords ('/tmp/cords.'+uniqID)
  LET borderPartNos ('/tmp/borderPartNos.'+uniqID)
  LET Triangles ('/tmp/trianglePartNos.'+uniqID)
  LET blankInfos ('/tmp/blankInfos.'+uniqID)


  RUN GRAPHIC ('date > ' + blockTimer +'; rm -f '+xys)

{
  OPEN_INFILE 9 ('| grep "UNBLENDED FILLET RADII MAY BE BLENDED 0.25 mm MAXIMUM"
/tmp/mdx/'+sub_filename)
  READ_FILE 9 olev1
  CLOSE_FILE 9
  OPEN_INFILE 9 ('| grep "2Dseed" /tmp/mdx/'+sub_filename)
  READ_FILE 9 olev2
  CLOSE_FILE 9
  IF ((olev1 <> 'END-OF-FILE') AND (olev2 <> 'END-OF-FILE'))
    LET olev "olev"
  ELSE
    LET olev "END-OF-FILE"
  END_IF
}
```

```
{before tagging infos, remove all mkm infos in the file}
  RUN GRAPHIC ('/usr/bin/grep mkm '+ TMPMDX+"/"+sub_filename + ' > '
+currentInfos)
  OPEN_INFILE 2 currentInfos
  READ_FILE 2 infoLine
  WHILE (infoLine <> 'END-OF-FILE')
    CHANGE_GLOBAL_INFO infoLine ''
    READ_FILE 2 infoLine
  END_WHILE
  CLOSE_FILE 2

  LET mysheetnum 0
  {PURGE_FILE message_file CONFIRM}

  UNITS MM
  CATCH NO_VIEWPORT_RANGE 1

  OPEN_OUTFILE 3 APPEND xys

  EDIT_PART TOP
  PARTS_LIST TREE DEL_OLD Input_file

  {===GET BORDER PARTS SHEET SIZE AND UNIQ ID===}
  {RUN GRAPHIC (mdxbin+'/getDrawingPartNos '+Input_file+' > '+borderPartNos)}
  RUN GRAPHIC (MDXDIR+'/bin/mygetParentChild.new '+Input_file+' HP_TEXT >
'+borderPartNos)

  {===FIND ANY REV_TRIANGLE PARTS===}
  RUN GRAPHIC (MDXDIR+'/bin/getParentChild '+Input_file+' rev_triangle >
'+Triangles)

  {iterate through Border parts and map the RevBlocks and tag fields with infos}
  OPEN_INFILE 4 borderPartNos
  READ_FILE 4 myLine
  {get the first character of myline, if its NOT ~ then continue else nextline}
  LET firstChar (SUBSTR myLine 1 1)

    WHILE (myLine<>'END-OF-FILE')                 {READ FROM PARTLIST FILE TILL
EOF}
 IF (mmd="ON") DISPLAY ("LINE FROM PARTLIST:" + myline) END_IF
      IF (firstChar <>'~')                        {IF MYLINE STARTS WITH A, B, C,
D, E}
        CATCH NO_VIEWPORT_RANGE 1                 {ITS A BORDER... PROCESS IT}
        LET mysheetnum (mysheetnum +1)            {OTHERWISE ITS THE UNIQ ID OF A
CHILD}
        OPEN_OUTFILE 5 DEL_OLD message_file
        LET space (POS myLine ' ')
        LET bd_size (SUBSTR myLine 1 (space -1))
        LET Border_ID (SUBSTR myLine (space +1) 80)
        EDIT_PART Border_ID
        INQ_ENV 7
        LET LL (INQ 101)
        LET UR (INQ 102)
        WINDOW LL UR
 IF (mmd="ON") DISPLAY ("border size is: " + STR bd_size) END_IF
        LET Oborder_width ((X_OF UR) - (X_OF LL))
 IF (mmd="ON")
   DISPLAY ("border width is: *" +STR Oborder_width+"*")
   DISPLAY ("Border width is: " + STR Oborder_width)
 END_IF
```

```
        LET border_width (SUBSTR (STR Oborder_width) 1 3)
        IF (mmd="ON") DISPLAY ("Border width is: " + STR border_width) END_IF


{======MAIN======}
        IF ((STR border_width = "266") OR (STR border_width = "390") OR (STR
border_width = "523") OR (STR border_width = "524") OR (STR border_width =
"820") OR (STR border_width = "821") OR (STR border_width ="107"))
            LET border_type "OLD"
IF (mmd="ON") DISPLAY ("this is an OLD border") END_IF
            find_border_type 'x'
            find_border_type 'y'
            EDIT_PART Border_ID
            tagrb
            tagtb
        ELSE
            LET border_type "NEW"
IF (mmd="ON") DISPLAY ("this is a NEW Structured border") END_IF
            EDIT_PART Border_ID
            newTagTB
            newTagNotes
            newTagRB
            IF (rev_type = "A")
              newDelRB
            END_IF
        END_IF
{===END=MAIN=====}
    END_IF
    READ_FILE 4 myLine
    LET firstChar (SUBSTR myLine 1 1)
  END_WHILE
  CLOSE_FILE 4

  RUN GRAPHIC (MDXDIR+'/bin/rbmap '+ xys +'>'+ cords)
  {tagblanks}

  IF ((rev_type = "A") AND (border_type = "OLD"))
    delrb

    OPEN_INFILE 6 Triangles
    READ_FILE 6 parentChild
    WHILE (parentChild <> 'END-OF-FILE')
      LET space (POS parentChild ' ')
      LET myParent (SUBSTR parentChild 1 (space -1))
      LET myChild (SUBSTR parentChild (space +1) 80)
      EDIT_PART myParent
      DELETE myChild
      READ_FILE 6 parentChild
    END_WHILE
    CLOSE_FILE 6
    END
  END_IF

  IF (rev_type = "B")
    {I think the following needs of the A+ revs will be done in medex:}
    {find at least one revB triangle }
    {must have PCO number in RevDescription field }
    {only the rev description of the FIRST LINE of the rev block records is
  tagged}
    {on Xrevs the revDescription can default to X REVISION UPDATE}
    {on B+ revs I need to tag the last line of the revBlock??}
  END_IF
```

```
   proprietary_check uniqID

   CLOSE_FILE 3
   CLOSE_FILE 5
   END

   STORE MI ALL DEL_OLD orgFile CONFIRM

{### This section of code will run inqmi on the newFile using all the possible
infos
     after finding out how many sheets are in the drawing. Any "info not found"
from
     inqmi will get listed. This list will be used by the continuation of this
macro
     to add text and tag with infos the pieces of text that are missing.
}

{===THIS CODE IS NOT USED===
   OPEN_INFILE 1 ('| '+MDXDIR+'/bin/getpartlist '+orgFile+' | grep ' +'"[A-
E]_.*_DRAWING" | wc -l')
   READ_FILE 1 mysheets
   CLOSE_FILE 1
   RUN GRAPHIC (MDXDIR+'/bin/checkBlankInfos '+mysheets+' '+orgFile+' '+uniqID+'
| sort > '+blankInfos)
   OPEN_INFILE 1 ('| wc -l '+blankInfos+' | cut -d" " -f1')
   READ_FILE 1 blanks
   CLOSE_FILE 1

   IF (blanks <> 0)
     DISPLAY ("Number of Blank Infos is: " + blanks)
     addTextElements
   END IF
}

   DACA
   RUN GRAPHIC ('chmod 655 '+orgFile+';mv '+orgFile+' '+newFile+';rm -f
/tmp/*'+uniqID+'* '+TMPMDX+'/'+sub_filename)

   RUN GRAPHIC (MDXDIR+'/bin/getInfoValues '+newFile+' sdf 1 > '+TMPMDX+'/infos')

   RUN GRAPHIC ('date >> ' + blockTimer)
END_DEFINE
{
DwgNumberU
Er
Rev
3dRev
RevDesc
ApprovedBy
ApprovedDate
DrawnBy
DrawnDate
EngChecker
CheckDate
RelToProd
RelDate
FileRevisedBy
FileRevisedByDate
Title
Title
```

```
PartNumber
Scale
SheetNo
SheetQty
DwgNumberL
Notes
Item
ItemQty
PartMatlDescription
MatlPartNo
MatlDwgNo
MatlSpec
}
```

```
DEFINE proprietary_check
  PARAMETER uniqID
  LOCAL Input_file
  LOCAL delpart
  LOCAL myparent
  LOCAL mychild
  LOCAL resstamp
  LOCAL myLL

  LET resstamp (MDXDIR+'/data/restricted_stamp')
  LET Input_file ('/tmp/Me30_parts_list.'+uniqID)

  PARTS_LIST TREE DEL_OLD Input_file
  OPEN_INFILE 1 ('| '+MDXDIR+'/bin/getParentChild '+Input_file+'
PROPRIETARY_STAMP')
  READ_FILE 1 delpart
  WHILE (delpart <> 'END-OF-FILE')
    LET delim (POS delpart ' ')
    LET myparent (SUBSTR delpart 1 (delim -1))
    LET mychild  (SUBSTR delpart (delim +1) 500)
    EDIT_PART mychild
    INQ_ENV 7
    LET myLL (INQ 101)
    EDIT_PART myparent
    DELETE mychild
    LOAD SUBPART resstamp myLL
    READ_FILE 1 delpart
  END_WHILE
END_DEFINE
```

```
DEFINE tagtb

  LOCAL rowNum

  LET rowNum 1
IF (bd_size = 'E')
  LET drawnByLL ((PNT_XY 867.488434177725 31.2603631144714) +LL)
  LET drawnByUR ((PNT_XY 914.46939078567 40.9329130043424) +LL)
  LET mid_X (((X_OF drawnByLL) + (X_OF drawnByUR)) / 2)
  LET mid_Y (((Y_OF drawnByLL) + (Y_OF drawnByUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmDrawnBy")
  tag InfoText drawnByLL drawnByUR midpoint

  LET drawnBy_dateLL ((PNT_XY 914.46939078567 31.5367216827535) + LL)
  LET drawnBy_dateUR ((PNT_XY 939.618020499335 40.9329130043424) + LL)
  LET mid_X (((X_OF drawnBy_dateLL) + (X_OF drawnBy_dateUR)) / 2)
  LET mid_Y (((Y_OF drawnBy_dateLL) + (Y_OF drawnBy_dateUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmDrawnDate")
  tag InfoText drawnBy_dateLL drawnBy_dateUR midpoint

  LET engrCheckerLL ((PNT_XY 867.212075609443 21.0350960880364) + LL)
  LET engrCheckerUR ((PNT_XY 914.745749353952 30.9840045461894) + LL)
  LET mid_X (((X_OF engrCheckerLL) + (X_OF engrCheckerUR)) / 2)
  LET mid_Y (((Y_OF engrCheckerLL) + (Y_OF engrCheckerUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmEngChecker")
  tag InfoText engrCheckerLL engrCheckerUR midpoint

  LET engrChecker_dateLL ((PNT_XY 915.022107922234 21.3114546563184) + LL)
  LET engrChecker_dateUR ((PNT_XY 939.894379067617 30.9840045461894) + LL)
  LET mid_X (((X_OF engrChecker_dateLL) + (X_OF engrChecker_dateUR)) / 2)
  LET mid_Y (((Y_OF engrChecker_dateLL) + (Y_OF engrChecker_dateUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmCheckDate")
  tag InfoText engrChecker_dateLL engrChecker_dateUR midpoint

  LET relToProdLL ((PNT_XY 867.764792746007 11.6389047664475) + LL)
  LET relToProdUR ((PNT_XY 914.745749353952 21.0350960880364) + LL)
  LET mid_X (((X_OF relToProdLL) + (X_OF relToProdUR)) / 2)
  LET mid_Y (((Y_OF relToProdLL) + (Y_OF relToProdUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmRelToProd")
  tag InfoText relToProdLL relToProdUR midpoint

  LET relToProd_dateLL ((PNT_XY 914.46939078567 11.6389047664475) + LL)
  LET relToProd_dateUR ((PNT_XY 940.170737635899 20.4823789514724) + LL)
  LET mid_X (((X_OF relToProd_dateLL) + (X_OF relToProd_dateUR)) / 2)
  LET mid_Y (((Y_OF relToProd_dateLL) + (Y_OF relToProd_dateUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmRelDate")
  tag InfoText relToProd_dateLL relToProd_dateUR midpoint

  LET fileRevisedByLL ((PNT_XY 867.764792746007 0.0318448986023299) + LL)
  LET fileRevisedByUR ((PNT_XY 914.46939078567 11.3625461981655) + LL)
  LET mid_X (((X_OF fileRevisedByLL) + (X_OF fileRevisedByUR)) / 2)
  LET mid_Y (((Y_OF fileRevisedByLL) + (Y_OF fileRevisedByUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
```

```
LET InfoText (STR mysheetnum + "_mkmFileRevisedBy")
tag InfoText fileRevisedByLL fileRevisedByUR midpoint

LET fileRevisedBy_dateLL ((PNT_XY 914.745749353952 0.0318448986023299) + LL)
LET fileRevisedBy_dateUR ((PNT_XY 939.618020499335 11.0861876298834) + LL)
LET mid_X (((X_OF fileRevisedBy_dateLL) + (X_OF fileRevisedBy_dateUR)) / 2)
LET mid_Y (((Y_OF fileRevisedBy_dateLL) + (Y_OF fileRevisedBy_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedByDate")
tag InfoText fileRevisedBy_dateLL fileRevisedBy_dateUR midpoint

LET dwgTitleLL ((PNT_XY 939.618020499335 11.6389047664475) + LL)
LET dwgTitleUR ((PNT_XY 1007.878586865 40.9329130043424) + LL)
LET mid_X (((X_OF dwgTitleLL) + (X_OF dwgTitleUR)) / 2)
LET mid_Y (((Y_OF dwgTitleLL) + (Y_OF dwgTitleUR)) / 2)
{ LET mid_Y ((Y_OF dwgTitleLL) + 4))}
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmTitle")
tag InfoText dwgTitleLL dwgTitleUR midpoint

LET dwgPartNumberLL ((PNT_XY 1008.43130400156 11.6389047664475) + LL)
LET dwgPartNumberUR ((PNT_XY 1072.27013327471 21.3114546563184) + LL)
LET mid_X (((X_OF dwgPartNumberLL) + (X_OF dwgPartNumberUR)) / 2)
LET mid_Y (((Y_OF dwgPartNumberLL) + (Y_OF dwgPartNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmPartNumber")
tag InfoText dwgPartNumberLL dwgPartNumberUR midpoint

LET dwgScaleLL ((PNT_XY 939.894379067617 0.308203466884365) + LL)
LET dwgScaleUR ((PNT_XY 964.766650212999 11.0861876298834) + LL)
LET mid_X (((X_OF dwgScaleLL) + (X_OF dwgScaleUR)) / 2)
LET mid_Y (((Y_OF dwgScaleLL) + (Y_OF dwgScaleUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmScale")
tag InfoText dwgScaleLL dwgScaleUR midpoint

LET sheetNumberLL ((PNT_XY 965.043008781281 0.0318448986023299) + LL)
LET sheetNumberUR ((PNT_XY 984.111749992741 11.3625461981655) + LL)
LET mid_X (((X_OF sheetNumberLL) + (X_OF sheetNumberUR)) / 2)
LET mid_Y (((Y_OF sheetNumberLL) + (Y_OF sheetNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetNo")
tag InfoText sheetNumberLL sheetNumberUR midpoint

LET numberOfSheetsLL ((PNT_XY 984.111749992741 0.308203466884365) + LL)
LET numberOfSheetsUR ((PNT_XY 998.482395543406 11.0861876298834) + LL)
LET mid_X (((X_OF numberOfSheetsLL) + (X_OF numberOfSheetsUR)) / 2)
LET mid_Y (((Y_OF numberOfSheetsLL) + (Y_OF numberOfSheetsUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetQty")
tag InfoText numberOfSheetsLL numberOfSheetsUR midpoint

LET drawingNumberLL ((PNT_XY 998.758754111689 0.0318448986023299) + LL)
LET drawingNumberUR ((PNT_XY 1071.99377470643 11.6389047664475) + LL)
LET mid_X (((X_OF drawingNumberLL) + (X_OF drawingNumberUR)) / 2)
LET mid_Y (((Y_OF drawingNumberLL) + (Y_OF drawingNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmDwgNumberL")
tag InfoText drawingNumberLL drawingNumberUR midpoint

LET itemNumberLL ((PNT_XY 867.556912407033 46.3806007109646) + LL)
```

```
   LET itemNumberUR ((PNT_XY 880.518862954774 54.2066840605444) + LL)
   LET mid_X (((X_OF itemNumberLL) + (X_OF itemNumberUR)) / 2)
   LET mid_Y (((Y_OF itemNumberLL) + (Y_OF itemNumberUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkm_B"+STR rowNum+"Item")
   tag InfoText itemNumberLL itemNumberUR midpoint

   LET itemQuantityLL ((PNT_XY 880.518862954774 46.625165815639) + LL)
   LET itemQuantityUR ((PNT_XY 892.747118188493 53.96211895587) + LL)
   LET mid_X (((X_OF itemQuantityLL) + (X_OF itemQuantityUR)) / 2)
   LET mid_Y (((Y_OF itemQuantityLL) + (Y_OF itemQuantityUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkm_B"+STR rowNum+ "ItemQty")
   tag InfoText itemQuantityLL itemQuantityUR midpoint

   LET partMatDescriptionLL ((PNT_XY 892.257987979144 46.625165815639) + LL)
   LET partMatDescriptionUR ((PNT_XY 972.964472521685 54.2066840605444) + LL)
   LET mid_X (((X_OF partMatDescriptionLL) + (X_OF partMatDescriptionUR)) / 2)
   LET mid_Y (((Y_OF partMatDescriptionLL) + (Y_OF partMatDescriptionUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkm_B"+STR rowNum+"PartMatlDescription")
   tag InfoText partMatDescriptionLL partMatDescriptionUR midpoint

   LET materialPartNumberLL ((PNT_XY 972.964472521685 46.625165815639) + LL)
   LET materialPartNumberUR ((PNT_XY 1008.18184759479 53.96211895587) + LL)
   LET mid_X (((X_OF materialPartNumberLL) + (X_OF materialPartNumberUR)) / 2)
{  LET mid_Y (((Y_OF materialPartNumberLL) + (Y_OF materialPartNumberUR)) / 2) }
   LET mid_Y ((Y_OF materialPartNumberLL) + 2))
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkm_B" +STR rowNum+ "MatlPartNo")
   tag InfoText materialPartNumberLL materialPartNumberUR midpoint

   LET materialDrawingNumberLL ((PNT_XY 1008.42641269947 46.3806007109646) + LL)
   LET materialDrawingNumberUR ((PNT_XY 1040.46444141181 53.96211895587) + LL)
   LET mid_X (((X_OF materialDrawingNumberLL) + (X_OF materialDrawingNumberUR)) /
2)
{  LET mid_Y (((Y_OF materialDrawingNumberLL) + (Y_OF materialDrawingNumberUR))
/ 2) }
   LET mid_Y ((Y_OF materialDrawingNumberLL) + 2))
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkm_B"+ STR rowNum+ "MatlDwgNo")
   tag InfoText materialDrawingNumberLL materialDrawingNumberUR midpoint

   LET materialSpecLL ((PNT_XY 1040.21987630714 46.625165815639) + LL)
   LET materialSpecUR ((PNT_XY 1072.0133399148 54.4512491652187) + LL)
   LET mid_X (((X_OF materialSpecLL) + (X_OF materialSpecUR)) / 2)
{  LET mid_Y (((Y_OF materialSpecLL) + (Y_OF materialSpecUR)) / 2) }
   LET mid_Y ((Y_OF materialSpecLL) + 2))
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkm_B"+ STR rowNum+ "MatlSpec")
   tag InfoText materialSpecLL materialSpecUR midpoint

ELSE_IF (bd_size = 'D')
   LET drawnByLL ((PNT_XY 616.906540125519 31.4277429577606) + LL)
   LET drawnByUR ((PNT_XY 663.16640199767 41.0152790970665) + LL)
   LET mid_X (((X_OF drawnByLL) + (X_OF drawnByUR)) / 2)
   LET mid_Y (((Y_OF drawnByLL) + (Y_OF drawnByUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmDrawnBy")
   tag InfoText drawnByLL drawnByUR midpoint
```

```
LET drawnBy_dateLL ((PNT_XY 662.926713594188 31.6674313612433) + LL)
LET drawnBy_dateUR ((PNT_XY 688.573372766831 40.5359022901012) + LL)
LET mid_X (((X_OF drawnBy_dateLL) + (X_OF drawnBy_dateUR)) / 2)
LET mid_Y (((Y_OF drawnBy_dateLL) + (Y_OF drawnBy_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmDrawnDate")
tag InfoText drawnBy_dateLL drawnBy_dateUR midpoint

LET engrCheckerLL ((PNT_XY 616.666851722037 21.1211416080068) + LL)
LET engrCheckerUR ((PNT_XY 662.926713594188 31.188054554278) + LL)
LET mid_X (((X_OF engrCheckerLL) + (X_OF engrCheckerUR)) / 2)
LET mid_Y (((Y_OF engrCheckerLL) + (Y_OF engrCheckerUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmEngChecker")
tag InfoText engrCheckerLL engrCheckerUR midpoint

LET engrChecker_dateLL ((PNT_XY 663.16640199767 21.3608300114894) + LL)
LET engrChecker_dateUR ((PNT_XY 688.333684363348 31.188054554278) + LL)
LET mid_X (((X_OF engrChecker_dateLL) + (X_OF engrChecker_dateUR)) / 2)
LET mid_Y (((Y_OF engrChecker_dateLL) + (Y_OF engrChecker_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmCheckDate")
tag InfoText engrChecker_dateLL engrChecker_dateUR midpoint

LET relToProdLL ((PNT_XY 616.906540125519 11.2939170652182) + LL)
LET relToProdUR ((PNT_XY 663.16640199767 20.6417648010415) + LL)
LET mid_X (((X_OF relToProdLL) + (X_OF relToProdUR)) / 2)
LET mid_Y (((Y_OF relToProdLL) + (Y_OF relToProdUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmRelToProd")
tag InfoText relToProdLL relToProdUR midpoint

LET relToProd_dateLL ((PNT_XY 662.926713594188 11.2939170652182) + LL)
LET relToProd_dateUR ((PNT_XY 688.333684363348 20.6417648010415) + LL)
LET mid_X (((X_OF relToProd_dateLL) + (X_OF relToProd_dateUR)) / 2)
LET mid_Y (((Y_OF relToProd_dateLL) + (Y_OF relToProd_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmRelDate")
tag InfoText relToProd_dateLL relToProd_dateUR midpoint

LET fileRevisedByLL ((PNT_XY 616.427163318554 0.268250505016397) + LL)
LET fileRevisedByUR ((PNT_XY 663.406090401153 11.2939170652182) + LL)
LET mid_X (((X_OF fileRevisedByLL) + (X_OF fileRevisedByUR)) / 2)
LET mid_Y (((Y_OF fileRevisedByLL) + (Y_OF fileRevisedByUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedBy")
tag InfoText fileRevisedByLL fileRevisedByUR midpoint

LET fileRevisedBy_dateLL ((PNT_XY 662.926713594188 0.268250505016397) + LL)
LET fileRevisedBy_dateUR ((PNT_XY 688.333684363348 11.2939170652182) + LL)
LET mid_X (((X_OF fileRevisedBy_dateLL) + (X_OF fileRevisedBy_dateUR)) / 2)
LET mid_Y (((Y_OF fileRevisedBy_dateLL) + (Y_OF fileRevisedBy_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedByDate")
tag InfoText fileRevisedBy_dateLL fileRevisedBy_dateUR midpoint

LET dwgTitleLL ((PNT_XY 688.333684363348 11.2939170652182) + LL)
LET dwgTitleUR ((PNT_XY 756.884567759386 41.2549675005492) + LL)
LET mid_X (((X_OF dwgTitleLL) + (X_OF dwgTitleUR)) / 2)
LET mid_Y (((Y_OF dwgTitleLL) + (Y_OF dwgTitleUR)) / 2)
{   LET mid_Y ((Y_OF dwgTitleLL) + 4))}
```

```
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmTitle")
tag InfoText dwgTitleLL dwgTitleUR midpoint

LET dwgPartNumberLL ((PNT_XY 756.884567759386 11.7732938721835) + LL)
LET dwgPartNumberUR ((PNT_XY 820.881371489253 21.1211416080068) + LL)
LET mid_X (((X_OF dwgPartNumberLL) + (X_OF dwgPartNumberUR)) / 2)
LET mid_Y (((Y_OF dwgPartNumberLL) + (Y_OF dwgPartNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmPartNumber")
tag InfoText dwgPartNumberLL dwgPartNumberUR midpoint

LET dwgScaleLL ((PNT_XY 688.573372766831 0.0285621015337512) + LL)
LET dwgScaleUR ((PNT_XY 713.740655132509 11.2939170652182) + LL)
LET mid_X (((X_OF dwgScaleLL) + (X_OF dwgScaleUR)) / 2)
LET mid_Y (((Y_OF dwgScaleLL) + (Y_OF dwgScaleUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmScale")
tag InfoText dwgScaleLL dwgScaleUR midpoint

LET sheetNumberLL ((PNT_XY 713.740655132509 0.0285621015337512) + LL)
LET sheetNumberUR ((PNT_XY 728.601336148433 11.2939170652182) + LL)
LET mid_X (((X_OF sheetNumberLL) + (X_OF sheetNumberUR)) / 2)
LET mid_Y (((Y_OF sheetNumberLL) + (Y_OF sheetNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetNo")
tag InfoText sheetNumberLL sheetNumberUR midpoint

LET numberOfSheetsLL ((PNT_XY 729.080712955399 -0.211126301948894) + LL)
LET numberOfSheetsUR ((PNT_XY 747.29703162008 11.0542286617356) + LL)
LET mid_X (((X_OF numberOfSheetsLL) + (X_OF numberOfSheetsUR)) / 2)
LET mid_Y (((Y_OF numberOfSheetsLL) + (Y_OF numberOfSheetsUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetQty")
tag InfoText numberOfSheetsLL numberOfSheetsUR midpoint

LET drawingNumberLL ((PNT_XY 747.29703162008 -0.211126301948894) + LL)
LET drawingNumberUR ((PNT_XY 820.881371489253 11.2939170652182) + LL)
LET mid_X (((X_OF drawingNumberLL) + (X_OF drawingNumberUR)) / 2)
LET mid_Y (((Y_OF drawingNumberLL) + (Y_OF drawingNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmDwgNumberL")
tag InfoText drawingNumberLL drawingNumberUR midpoint

LET itemNumberLL ((PNT_XY 616.322299642031 46.7809087402155) + LL)
LET itemNumberUR ((PNT_XY 629.417151247923 54.0558262990443) + LL)
LET mid_X (((X_OF itemNumberLL) + (X_OF itemNumberUR)) / 2)
LET mid_Y (((Y_OF itemNumberLL) + (Y_OF itemNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmItem")
tag InfoText itemNumberLL itemNumberUR midpoint

LET itemQuantityLL ((PNT_XY 629.174653995962 47.0234059921765) + LL)
LET itemQuantityUR ((PNT_XY 641.29951659401 54.0558262990443) + LL)
LET mid_X (((X_OF itemQuantityLL) + (X_OF itemQuantityUR)) / 2)
LET mid_Y (((Y_OF itemQuantityLL) + (Y_OF itemQuantityUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmItemQty")
tag InfoText itemQuantityLL itemQuantityUR midpoint

LET partMatDescriptionLL ((PNT_XY 641.542013845971 46.7809087402155) + LL)
```

```
    LET partMatDescriptionUR ((PNT_XY 721.566106993087 54.0558262990443) + LL)
    LET mid_X (((X_OF partMatDescriptionLL) + (X_OF partMatDescriptionUR)) / 2)
    LET mid_Y (((Y_OF partMatDescriptionLL) + (Y_OF partMatDescriptionUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmPartMatlDescription")
    tag InfoText partMatDescriptionLL partMatDescriptionUR midpoint

    LET materialPartNumberLL ((PNT_XY 721.808604245048 46.5384114882546) + LL)
    LET materialPartNumberUR ((PNT_XY 756.970705779388 53.5708317951224) + LL)
    LET mid_X (((X_OF materialPartNumberLL) + (X_OF materialPartNumberUR)) / 2)
{   LET mid_Y (((Y_OF materialPartNumberLL) + (Y_OF materialPartNumberUR)) / 2) }
    LET mid_Y ((Y_OF materialPartNumberLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmMatlPartNo")
    tag InfoText materialPartNumberLL materialPartNumberUR midpoint

    LET materialDrawingNumberLL ((PNT_XY 756.970705779388 46.7809087402155) + LL)
    LET materialDrawingNumberUR ((PNT_XY 789.222840290195 53.3283345431614) + LL)
    LET mid_X (((X_OF materialDrawingNumberLL) + (X_OF materialDrawingNumberUR)) /
2)
{   LET mid_Y (((Y_OF materialDrawingNumberLL) + (Y_OF materialDrawingNumberUR))
/ 2) }
    LET mid_Y ((Y_OF materialDrawingNumberLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmMatlDwgNo")
    tag InfoText materialDrawingNumberLL materialDrawingNumberUR midpoint

    LET materialSpecLL ((PNT_XY 789.222840290195 46.5384114882546) + LL)
    LET materialSpecUR ((PNT_XY 820.504985793159 53.8133290470834) + LL)
    LET mid_X (((X_OF materialSpecLL) + (X_OF materialSpecUR)) / 2)
{   LET mid_Y (((Y_OF materialSpecLL) + (Y_OF materialSpecUR)) / 2) }
    LET mid_Y ((Y_OF materialSpecLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmMatlSpec")
    tag InfoText materialSpecLL materialSpecUR midpoint

ELSE_IF (bd_size = 'C')
    LET drawnByLL ((PNT_XY 320.088651946856 31.2558158876063) +LL)
    LET drawnByUR ((PNT_XY 366.711156416308 40.6699754439379) +LL)
    LET mid_X (((X_OF drawnByLL) + (X_OF drawnByUR)) / 2)
    LET mid_Y (((Y_OF drawnByLL) + (Y_OF drawnByUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmDrawnBy")
    tag InfoText drawnByLL drawnByUR midpoint

    LET drawnBy_dateLL ((PNT_XY 366.711156416308 31.2558158876063) + LL)
    LET drawnBy_dateUR ((PNT_XY 391.591435243756 40.8941221000411) + LL)
    LET mid_X (((X_OF drawnBy_dateLL) + (X_OF drawnBy_dateUR)) / 2)
    LET mid_Y (((Y_OF drawnBy_dateLL) + (Y_OF drawnBy_dateUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmDrawnDate")
    tag InfoText drawnBy_dateLL drawnBy_dateUR midpoint

    LET engrCheckerLL ((PNT_XY 319.864505290753 20.945069706862) + LL)
    LET engrCheckerUR ((PNT_XY 366.487009760205 31.2558158876063) + LL)
    LET mid_X (((X_OF engrCheckerLL) + (X_OF engrCheckerUR)) / 2)
    LET mid_Y (((Y_OF engrCheckerLL) + (Y_OF engrCheckerUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmEngChecker")
    tag InfoText engrCheckerLL engrCheckerUR midpoint
```

```
LET engrChecker_dateLL ((PNT_XY 366.487009760205 21.1692163629652) + LL)
LET engrChecker_dateUR ((PNT_XY 391.591435243756 31.2558158876063) + LL)
LET mid_X (((X_OF engrChecker_dateLL) + (X_OF engrChecker_dateUR)) / 2)
LET mid_Y (((Y_OF engrChecker_dateLL) + (Y_OF engrChecker_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmCheckDate")
tag InfoText engrChecker_dateLL engrChecker_dateUR midpoint

LET relToProdLL ((PNT_XY 319.864505290753 11.5309101505303) + LL)
LET relToProdUR ((PNT_XY 366.711156416308 20.945069706862) + LL)
LET mid_X (((X_OF relToProdLL) + (X_OF relToProdUR)) / 2)
LET mid_Y (((Y_OF relToProdLL) + (Y_OF relToProdUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmRelToProd")
tag InfoText relToProdLL relToProdUR midpoint

LET relToProd_dateLL ((PNT_XY 366.487009760205 11.7550568066335) + LL)
LET relToProd_dateUR ((PNT_XY 391.591435243756 20.7209230507589) + LL)
LET mid_X (((X_OF relToProd_dateLL) + (X_OF relToProd_dateUR)) / 2)
LET mid_Y (((Y_OF relToProd_dateLL) + (Y_OF relToProd_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmRelDate")
tag InfoText relToProd_dateLL relToProd_dateUR midpoint

LET fileRevisedByLL ((PNT_XY 319.640358634649 0.0994306892704202) + LL)
LET fileRevisedByUR ((PNT_XY 366.711156416308 11.7550568066335) + LL)
LET mid_X (((X_OF fileRevisedByLL) + (X_OF fileRevisedByUR)) / 2)
LET mid_Y (((Y_OF fileRevisedByLL) + (Y_OF fileRevisedByUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedBy")
tag InfoText fileRevisedByLL fileRevisedByUR midpoint

LET fileRevisedBy_dateLL ((PNT_XY 366.262863104102 0.547724001476695) + LL)
LET fileRevisedBy_dateUR ((PNT_XY 391.815581899859 11.5309101505303) + LL)
LET mid_X (((X_OF fileRevisedBy_dateLL) + (X_OF fileRevisedBy_dateUR)) / 2)
LET mid_Y (((Y_OF fileRevisedBy_dateLL) + (Y_OF fileRevisedBy_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedByDate")
tag InfoText fileRevisedBy_dateLL fileRevisedBy_dateUR midpoint

LET dwgTitleLL ((PNT_XY 391.815581899859 11.5309101505303) + LL)
LET dwgTitleUR ((PNT_XY 459.956165355212 41.1182687561442) + LL)
LET mid_X (((X_OF dwgTitleLL) + (X_OF dwgTitleUR)) / 2)
LET mid_Y (((Y_OF dwgTitleLL) + (Y_OF dwgTitleUR)) / 2)
{   LET mid_Y ((Y_OF dwgTitleLL) + 4))}
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmTitle")
tag InfoText dwgTitleLL dwgTitleUR midpoint

LET dwgPartNumberLL ((PNT_XY 460.180312011315 11.7550568066335) + LL)
LET dwgPartNumberUR ((PNT_XY 523.837962344606 21.1692163629652) + LL)
LET mid_X (((X_OF dwgPartNumberLL) + (X_OF dwgPartNumberUR)) / 2)
LET mid_Y (((Y_OF dwgPartNumberLL) + (Y_OF dwgPartNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmPartNumber")
tag InfoText dwgPartNumberLL dwgPartNumberUR midpoint

LET dwgScaleLL ((PNT_XY 392.039728555962 0.32357734537355) + LL)
LET dwgScaleUR ((PNT_XY 416.92000738341 10.8584701822209) + LL)
LET mid_X (((X_OF dwgScaleLL) + (X_OF dwgScaleUR)) / 2)
LET mid_Y (((Y_OF dwgScaleLL) + (Y_OF dwgScaleUR)) / 2)
```

```
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmScale")
tag InfoText dwgScaleLL dwgScaleUR midpoint

LET sheetNumberLL ((PNT_XY 416.92000738341 0.0994306892704202) + LL)
LET sheetNumberUR ((PNT_XY 432.161979998423 11.5309101505303) + LL)
LET mid_X (((X_OF sheetNumberLL) + (X_OF sheetNumberUR)) / 2)
LET mid_Y (((Y_OF sheetNumberLL) + (Y_OF sheetNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetNo")
tag InfoText sheetNumberLL sheetNumberUR midpoint

LET numberOfSheetsLL ((PNT_XY 431.93783334232 -0.348862622935854) + LL)
LET numberOfSheetsUR ((PNT_XY 450.542005798881 11.3067634944272) + LL)
LET mid_X (((X_OF numberOfSheetsLL) + (X_OF numberOfSheetsUR)) / 2)
LET mid_Y (((Y_OF numberOfSheetsLL) + (Y_OF numberOfSheetsUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetQty")
tag InfoText numberOfSheetsLL numberOfSheetsUR midpoint

LET drawingNumberLL ((PNT_XY 450.766152454984 -0.12471596683271) + LL)
LET drawingNumberUR ((PNT_XY 524.062109000709 11.5309101505303) + LL)
LET mid_X (((X_OF drawingNumberLL) + (X_OF drawingNumberUR)) / 2)
LET mid_Y (((Y_OF drawingNumberLL) + (Y_OF drawingNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmDwgNumberL")
tag InfoText drawingNumberLL drawingNumberUR midpoint

LET itemNumberLL ((PNT_XY 319.816687337451 46.4039457692643) + LL)
LET itemNumberUR ((PNT_XY 332.203330415918 54.116383912461) + LL)
LET mid_X (((X_OF itemNumberLL) + (X_OF itemNumberUR)) / 2)
LET mid_Y (((Y_OF itemNumberLL) + (Y_OF itemNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmItem")
tag InfoText itemNumberLL itemNumberUR midpoint

LET itemQuantityLL ((PNT_XY 332.203330415918 46.6376560160278) + LL)
LET itemQuantityUR ((PNT_XY 344.589973494386 53.8826736656974) + LL)
LET mid_X (((X_OF itemQuantityLL) + (X_OF itemQuantityUR)) / 2)
LET mid_Y (((Y_OF itemQuantityLL) + (Y_OF itemQuantityUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmItemQty")
tag InfoText itemQuantityLL itemQuantityUR midpoint

LET partMatDescriptionLL ((PNT_XY 344.589973494386 46.4039457692643) + LL)
LET partMatDescriptionUR ((PNT_XY 425.220008627805 54.116383912461) + LL)
LET mid_X (((X_OF partMatDescriptionLL) + (X_OF partMatDescriptionUR)) / 2)
LET mid_Y (((Y_OF partMatDescriptionLL) + (Y_OF partMatDescriptionUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmPartMatlDescription")
tag InfoText partMatDescriptionLL partMatDescriptionUR midpoint

LET materialPartNumberLL ((PNT_XY 424.285167640751 46.4039457692643) + LL)
LET materialPartNumberUR ((PNT_XY 460.042835395572 54.3500941592245) + LL)
LET mid_X (((X_OF materialPartNumberLL) + (X_OF materialPartNumberUR)) / 2)
{ LET mid_Y (((Y_OF materialPartNumberLL) + (Y_OF materialPartNumberUR)) / 2) }
LET mid_Y ((Y_OF materialPartNumberLL) + 2))
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmMatlPartNo")
tag InfoText materialPartNumberLL materialPartNumberUR midpoint
```

```
   LET materialDrawingNumberLL ((PNT_XY 460.276545642336 46.4039457692643) + LL)
   LET materialDrawingNumberUR ((PNT_XY 492.762269942467 53.8826736656974) + LL)
   LET mid_X (((X_OF materialDrawingNumberLL) + (X_OF materialDrawingNumberUR)) /
2)
{  LET mid_Y (((Y_OF materialDrawingNumberLL) + (Y_OF materialDrawingNumberUR))
/ 2) }
   LET mid_Y ((Y_OF materialDrawingNumberLL) + 2))
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmMatlDwgNo")
   tag InfoText materialDrawingNumberLL materialDrawingNumberUR midpoint

   LET materialSpecLL ((PNT_XY 492.528559695704 46.4039457692643) + LL)
   LET materialSpecUR ((PNT_XY 524.079443008781 54.116383912461) + LL)
   LET mid_X (((X_OF materialSpecLL) + (X_OF materialSpecUR)) / 2)
{  LET mid_Y (((Y_OF materialSpecLL) + (Y_OF materialSpecUR)) / 2) }
   LET mid_Y ((Y_OF materialSpecLL) + 2))
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmMatlSpec")
   tag InfoText materialSpecLL materialSpecUR midpoint

ELSE_IF (bd_size = 'B')
   LET drawnByLL ((PNT_XY 185.687510629208 30.9076876651417) + LL)
   LET drawnByUR ((PNT_XY 232.679258216119 41.2011180889411) + LL)
   LET mid_X (((X_OF drawnByLL) + (X_OF drawnByUR)) / 2)
   LET mid_Y (((Y_OF drawnByLL) + (Y_OF drawnByUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmDrawnBy")
   tag InfoText drawnByLL drawnByUR midpoint

   LET drawnBy_dateLL ((PNT_XY 232.455487989514 31.3552281183503) + LL)
   LET drawnBy_dateUR ((PNT_XY 257.5177533692 40.9773478623368) + LL)
   LET mid_X (((X_OF drawnBy_dateLL) + (X_OF drawnBy_dateUR)) / 2)
   LET mid_Y (((Y_OF drawnBy_dateLL) + (Y_OF drawnBy_dateUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmDrawnDate")
   tag InfoText drawnBy_dateLL drawnBy_dateUR midpoint

   LET engrCheckerLL ((PNT_XY 185.911280855812 20.8380274679465) + LL)
   LET engrCheckerUR ((PNT_XY 232.23171776291 31.3552281183503) + LL)
   LET mid_X (((X_OF engrCheckerLL) + (X_OF engrCheckerUR)) / 2)
   LET mid_Y (((Y_OF engrCheckerLL) + (Y_OF engrCheckerUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmEngChecker")
   tag InfoText engrCheckerLL engrCheckerUR midpoint

   LET engrChecker_dateLL ((PNT_XY 232.455487989514 20.8380274679465) + LL)
   LET engrChecker_dateUR ((PNT_XY 257.741523595804 31.3552281183503) + LL)
   LET mid_X (((X_OF engrChecker_dateLL) + (X_OF engrChecker_dateUR)) / 2)
   LET mid_Y (((Y_OF engrChecker_dateLL) + (Y_OF engrChecker_dateUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmCheckDate")
   tag InfoText engrChecker_dateLL engrChecker_dateUR midpoint

   LET relToProdLL ((PNT_XY 186.135051082417 11.2159077239601) + LL)
   LET relToProdUR ((PNT_XY 232.23171776291 21.0617976945509) + LL)
   LET mid_X (((X_OF relToProdLL) + (X_OF relToProdUR)) / 2)
   LET mid_Y (((Y_OF relToProdLL) + (Y_OF relToProdUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmRelToProd")
   tag InfoText relToProdLL relToProdUR midpoint
```

```
LET relToProd_dateLL ((PNT_XY 232.455487989514 11.2159077239601) + LL)
LET relToProd_dateUR ((PNT_XY 257.741523595804 21.0617976945509) + LL)
LET mid_X (((X_OF relToProd_dateLL) + (X_OF relToProd_dateUR)) / 2)
LET mid_Y (((Y_OF relToProd_dateLL) + (Y_OF relToProd_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmRelDate")
tag InfoText relToProd_dateLL relToProd_dateUR midpoint

LET fileRevisedByLL ((PNT_XY 185.687510629208 0.0273963937433024) + LL)
LET fileRevisedByUR ((PNT_XY 232.23171776291 11.6634481771688) + LL)
LET mid_X (((X_OF fileRevisedByLL) + (X_OF fileRevisedByUR)) / 2)
LET mid_Y (((Y_OF fileRevisedByLL) + (Y_OF fileRevisedByUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedBy")
tag InfoText fileRevisedByLL fileRevisedByUR midpoint

LET fileRevisedBy_dateLL ((PNT_XY 232.455487989514 0.0273963937433024) + LL)
LET fileRevisedBy_dateUR ((PNT_XY 257.965293822409 11.4396779505644) + LL)
LET mid_X (((X_OF fileRevisedBy_dateLL) + (X_OF fileRevisedBy_dateUR)) / 2)
LET mid_Y (((Y_OF fileRevisedBy_dateLL) + (Y_OF fileRevisedBy_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedByDate")
tag InfoText fileRevisedBy_dateLL fileRevisedBy_dateUR midpoint

LET dwgTitleLL ((PNT_XY 257.741523595804 11.4396779505644) + LL)
LET dwgTitleUR ((PNT_XY 325.767672483522 40.7535776357325) + LL)
LET mid_X (((X_OF dwgTitleLL) + (X_OF dwgTitleUR)) / 2)
LET mid_Y (((Y_OF dwgTitleLL) + (Y_OF dwgTitleUR)) / 2)
{   LET mid_Y ((Y_OF dwgTitleLL) + 4))}
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmTitle")
tag InfoText dwgTitleLL dwgTitleUR midpoint

LET dwgPartNumberLL ((PNT_XY 326.215212936731 11.4396779505644) + LL)
LET dwgPartNumberUR ((PNT_XY 389.989727518967 21.5093381477595) + LL)
LET mid_X (((X_OF dwgPartNumberLL) + (X_OF dwgPartNumberUR)) / 2)
LET mid_Y (((Y_OF dwgPartNumberLL) + (Y_OF dwgPartNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmPartNumber")
tag InfoText dwgPartNumberLL dwgPartNumberUR midpoint

LET dwgScaleLL ((PNT_XY 257.5177533692 0.0273963937433024) + LL)
LET dwgScaleUR ((PNT_XY 282.80378897549 11.2159077239601) + LL)
LET mid_X (((X_OF dwgScaleLL) + (X_OF dwgScaleUR)) / 2)
LET mid_Y (((Y_OF dwgScaleLL) + (Y_OF dwgScaleUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmScale")
tag InfoText dwgScaleLL dwgScaleUR midpoint

LET sheetNumberLL ((PNT_XY 282.80378897549 0.0273963937433024) + LL)
LET sheetNumberUR ((PNT_XY 298.243934611189 11.2159077239601) + LL)
LET mid_X (((X_OF sheetNumberLL) + (X_OF sheetNumberUR)) / 2)
LET mid_Y (((Y_OF sheetNumberLL) + (Y_OF sheetNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetNo")
tag InfoText sheetNumberLL sheetNumberUR midpoint

LET numberOfSheetsLL ((PNT_XY 298.243934611189 0.474936846951962) + LL)
LET numberOfSheetsUR ((PNT_XY 316.593093192745 10.9921374973558) + LL)
LET mid_X (((X_OF numberOfSheetsLL) + (X_OF numberOfSheetsUR)) / 2)
LET mid_Y (((Y_OF numberOfSheetsLL) + (Y_OF numberOfSheetsUR)) / 2)
```

```
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetQty")
tag InfoText numberOfSheetsLL numberOfSheetsUR midpoint

LET drawingNumberLL ((PNT_XY 316.593093192745 -0.196373832861042) + LL)
LET drawingNumberUR ((PNT_XY 389.765957292363 11.2159077239601) + LL)
LET mid_X (((X_OF drawingNumberLL) + (X_OF drawingNumberUR)) / 2)
LET mid_Y (((Y_OF drawingNumberLL) + (Y_OF drawingNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmDwgNumberL")
tag InfoText drawingNumberLL drawingNumberUR midpoint

LET itemNumberLL ((PNT_XY 185.911280855812 46.1240630742365) + LL)
LET itemNumberUR ((PNT_XY 198.218643319051 54.1797912319926) + LL)
LET mid_X (((X_OF itemNumberLL) + (X_OF itemNumberUR)) / 2)
LET mid_Y (((Y_OF itemNumberLL) + (Y_OF itemNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmItem")
tag InfoText itemNumberLL itemNumberUR midpoint

LET itemQuantityLL ((PNT_XY 198.218643319051 46.3478333008409) + LL)
LET itemQuantityUR ((PNT_XY 210.749776008894 54.1797912319926) + LL)
LET mid_X (((X_OF itemQuantityLL) + (X_OF itemQuantityUR)) / 2)
LET mid_Y (((Y_OF itemQuantityLL) + (Y_OF itemQuantityUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmItemQty")
tag InfoText itemQuantityLL itemQuantityUR midpoint

LET partMatDescriptionLL ((PNT_XY 210.526005782289 46.1240630742365) + LL)
LET partMatDescriptionUR ((PNT_XY 291.08328735985 54.1797912319926) + LL)
LET mid_X (((X_OF partMatDescriptionLL) + (X_OF partMatDescriptionUR)) / 2)
LET mid_Y (((Y_OF partMatDescriptionLL) + (Y_OF partMatDescriptionUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmPartMatlDescription")
tag InfoText partMatDescriptionLL partMatDescriptionUR midpoint

LET materialPartNumberLL ((PNT_XY 290.859517133246 46.1240630742365) + LL)
LET materialPartNumberUR ((PNT_XY 325.991442710127 54.1797912319926) + LL)
LET mid_X (((X_OF materialPartNumberLL) + (X_OF materialPartNumberUR)) / 2)
{ LET mid_Y (((Y_OF materialPartNumberLL) + (Y_OF materialPartNumberUR)) / 2) }
LET mid_Y ((Y_OF materialPartNumberLL) + 2))
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmMatlPartNo")
tag InfoText materialPartNumberLL materialPartNumberUR midpoint

LET materialDrawingNumberLL ((PNT_XY 325.991442710127 46.1240630742365) + LL)
LET materialDrawingNumberUR ((PNT_XY 358.438125567755 53.9560210053883) + LL)
LET mid_X (((X_OF materialDrawingNumberLL) + (X_OF materialDrawingNumberUR)) /
2)
{ LET mid_Y (((Y_OF materialDrawingNumberLL) + (Y_OF materialDrawingNumberUR))
/ 2) }
LET mid_Y ((Y_OF materialDrawingNumberLL) + 2))
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmMatlDwgNo")
tag InfoText materialDrawingNumberLL materialDrawingNumberUR midpoint

LET materialSpecLL ((PNT_XY 357.990585114547 46.3478333008409) + LL)
LET materialSpecUR ((PNT_XY 389.989727518967 54.1797912319926) + LL)
LET mid_X (((X_OF materialSpecLL) + (X_OF materialSpecUR)) / 2)
{ LET mid_Y (((Y_OF materialSpecLL) + (Y_OF materialSpecUR)) / 2) }
LET mid_Y ((Y_OF materialSpecLL) + 2))
```

```
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmMatlSpec")
      tag InfoText materialSpecLL materialSpecUR midpoint

ELSE_IF (bd_size = 'A')
      LET drawnByLL ((PNT_XY 62.6256472409688 31.1835392254059) + LL)
      LET drawnByUR ((PNT_XY 109.261889333875 41.0488981296745) + LL)
      LET mid_X (((X_OF drawnByLL) + (X_OF drawnByUR)) / 2)
      LET mid_Y (((Y_OF drawnByLL) + (Y_OF drawnByUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmDrawnBy")
      tag InfoText drawnByLL drawnByUR midpoint

      LET drawnBy_dateLL ((PNT_XY 109.261889333875 31.6319646301453) + LL)
      LET drawnBy_dateUR ((PNT_XY 134.822137404026 41.0488981296745) + LL)
      LET mid_X (((X_OF drawnBy_dateLL) + (X_OF drawnBy_dateUR)) / 2)
      LET mid_Y (((Y_OF drawnBy_dateLL) + (Y_OF drawnBy_dateUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmDrawnDate")
      tag InfoText drawnBy_dateLL drawnBy_dateUR midpoint

      LET engrCheckerLL ((PNT_XY 62.6256472409688 20.8697549163977) + LL)
      LET engrCheckerUR ((PNT_XY 108.813463929136 31.1835392254059) + LL)
      LET mid_X (((X_OF engrCheckerLL) + (X_OF engrCheckerUR)) / 2)
      LET mid_Y (((Y_OF engrCheckerLL) + (Y_OF engrCheckerUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmEngChecker")
      tag InfoText engrCheckerLL engrCheckerUR midpoint

      LET engrChecker_dateLL ((PNT_XY 108.813463929136 21.0939676187674) + LL)
      LET engrChecker_dateUR ((PNT_XY 134.373711999286 31.4077519277756) + LL)
      LET mid_X (((X_OF engrChecker_dateLL) + (X_OF engrChecker_dateUR)) / 2)
      LET mid_Y (((Y_OF engrChecker_dateLL) + (Y_OF engrChecker_dateUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmCheckDate")
      tag InfoText engrChecker_dateLL engrChecker_dateUR midpoint

      LET relToProdLL ((PNT_XY 62.8498599433385 11.004396012129) + LL)
      LET relToProdUR ((PNT_XY 109.037676631506 21.3181803211372) + LL)
      LET mid_X (((X_OF relToProdLL) + (X_OF relToProdUR)) / 2)
      LET mid_Y (((Y_OF relToProdLL) + (Y_OF relToProdUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmRelToProd")
      tag InfoText relToProdLL relToProdUR midpoint

      LET relToProd_dateLL ((PNT_XY 109.261889333875 11.4528214168685) + LL)
      LET relToProd_dateUR ((PNT_XY 134.597924701656 20.645542214028) + LL)
      LET mid_X (((X_OF relToProd_dateLL) + (X_OF relToProd_dateUR)) / 2)
      LET mid_Y (((Y_OF relToProd_dateLL) + (Y_OF relToProd_dateUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmRelDate")
      tag InfoText relToProd_dateLL relToProd_dateUR midpoint

      LET fileRevisedByLL ((PNT_XY 62.1772218362293 0.242186298381355) + LL)
      LET fileRevisedByUR ((PNT_XY 109.037676631506 11.4528214168685) + LL)
      LET mid_X (((X_OF fileRevisedByLL) + (X_OF fileRevisedByUR)) / 2)
      LET mid_Y (((Y_OF fileRevisedByLL) + (Y_OF fileRevisedByUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      LET InfoText (STR mysheetnum + "_mkmFileRevisedBy")
      tag InfoText fileRevisedByLL fileRevisedByUR midpoint
```

```
LET fileRevisedBy_dateLL ((PNT_XY 108.589251226766 0.0179735960116147) + LL)
LET fileRevisedBy_dateUR ((PNT_XY 134.597924701656 11.2286087144987) + LL)
LET mid_X (((X_OF fileRevisedBy_dateLL) + (X_OF fileRevisedBy_dateUR)) / 2)
LET mid_Y (((Y_OF fileRevisedBy_dateLL) + (Y_OF fileRevisedBy_dateUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmFileRevisedByDate")
tag InfoText fileRevisedBy_dateLL fileRevisedBy_dateUR midpoint

LET dwgTitleLL ((PNT_XY 134.597924701656 11.4528214168685) + LL)
LET dwgTitleUR ((PNT_XY 202.982798924428 41.0488981296745) + LL)
LET mid_X (((X_OF dwgTitleLL) + (X_OF dwgTitleUR)) / 2)
LET mid_Y (((Y_OF dwgTitleLL) + (Y_OF dwgTitleUR)) / 2)
{  LET mid_Y ((Y_OF dwgTitleLL) + 4))}
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmTitle")
tag InfoText dwgTitleLL dwgTitleUR midpoint

LET dwgPartNumberLL ((PNT_XY 202.758586222058 11.6770341192382) + LL)
LET dwgPartNumberUR ((PNT_XY 266.883419099804 21.5423930235069) + LL)
LET mid_X (((X_OF dwgPartNumberLL) + (X_OF dwgPartNumberUR)) / 2)
LET mid_Y (((Y_OF dwgPartNumberLL) + (Y_OF dwgPartNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmPartNumber")
tag InfoText dwgPartNumberLL dwgPartNumberUR midpoint

LET dwgScaleLL ((PNT_XY 134.597924701656 0.0179735960116147) + LL)
LET dwgScaleUR ((PNT_XY 159.709747367067 11.2286087144987) + LL)
LET mid_X (((X_OF dwgScaleLL) + (X_OF dwgScaleUR)) / 2)
LET mid_Y (((Y_OF dwgScaleLL) + (Y_OF dwgScaleUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmScale")
tag InfoText dwgScaleLL dwgScaleUR midpoint

LET sheetNumberLL ((PNT_XY 159.261321962328 0.242186298381355) + LL)
LET sheetNumberUR ((PNT_XY 174.059360318731 11.4528214168685) + LL)
LET mid_X (((X_OF sheetNumberLL) + (X_OF sheetNumberUR)) / 2)
LET mid_Y (((Y_OF sheetNumberLL) + (Y_OF sheetNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetNo")
tag InfoText sheetNumberLL sheetNumberUR midpoint

LET numberOfSheetsLL ((PNT_XY 174.059360318731 0.242186298381355) + LL)
LET numberOfSheetsUR ((PNT_XY 193.117440020159 11.004396012129) + LL)
LET mid_X (((X_OF numberOfSheetsLL) + (X_OF numberOfSheetsUR)) / 2)
LET mid_Y (((Y_OF numberOfSheetsLL) + (Y_OF numberOfSheetsUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmSheetQty")
tag InfoText numberOfSheetsLL numberOfSheetsUR midpoint

LET drawingNumberLL ((PNT_XY 193.341652722529 0.242186298381355) + LL)
LET drawingNumberUR ((PNT_XY 266.434993695065 11.4528214168685) + LL)
LET mid_X (((X_OF drawingNumberLL) + (X_OF drawingNumberUR)) / 2)
LET mid_Y (((Y_OF drawingNumberLL) + (Y_OF drawingNumberUR)) / 2)
LET midpoint (PNT_XY mid_X mid_Y)
LET InfoText (STR mysheetnum + "_mkmDwgNumberL")
tag InfoText drawingNumberLL drawingNumberUR midpoint

LET itemNumberLL ((PNT_XY 62.6256472409688 46.2057902841786) + LL)
LET itemNumberUR ((PNT_XY 74.9573458713046 54.0532348671196) + LL)
LET mid_X (((X_OF itemNumberLL) + (X_OF itemNumberUR)) / 2)
LET mid_Y (((Y_OF itemNumberLL) + (Y_OF itemNumberUR)) / 2)
```

```
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmItem")
    tag InfoText itemNumberLL itemNumberUR midpoint

    LET itemQuantityLL ((PNT_XY 74.9573458713046 46.2057902841786) + LL)
    LET itemQuantityUR ((PNT_XY 87.2890445016405 54.0532348671196) + LL)
    LET mid_X (((X_OF itemQuantityLL) + (X_OF itemQuantityUR)) / 2)
    LET mid_Y (((Y_OF itemQuantityLL) + (Y_OF itemQuantityUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmItemQty")
    tag InfoText itemQuantityLL itemQuantityUR midpoint

    LET partMatDescriptionLL ((PNT_XY 87.5132572040102 46.4300029865484) + LL)
    LET partMatDescriptionUR ((PNT_XY 167.781404652378 54.0532348671196) + LL)
    LET mid_X (((X_OF partMatDescriptionLL) + (X_OF partMatDescriptionUR)) / 2)
    LET mid_Y (((Y_OF partMatDescriptionLL) + (Y_OF partMatDescriptionUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmPartMatlDescription")
    tag InfoText partMatDescriptionLL partMatDescriptionUR midpoint

    LET materialPartNumberLL ((PNT_XY 167.781404652378 46.4300029865484) + LL)
    LET materialPartNumberUR ((PNT_XY 202.758586222058 53.8290221647499) + LL)
    LET mid_X (((X_OF materialPartNumberLL) + (X_OF materialPartNumberUR)) / 2)
{   LET mid_Y (((Y_OF materialPartNumberLL) + (Y_OF materialPartNumberUR)) / 2) }
    LET mid_Y ((Y_OF materialPartNumberLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmMatlPartNo")
    tag InfoText materialPartNumberLL materialPartNumberUR midpoint

    LET materialDrawingNumberLL ((PNT_XY 203.207011626797 46.2057902841786) + LL)
    LET materialDrawingNumberUR ((PNT_XY 235.269428065671 54.0532348671196) + LL)
    LET mid_X (((X_OF materialDrawingNumberLL) + (X_OF materialDrawingNumberUR)) /
2)
{   LET mid_Y (((Y_OF materialDrawingNumberLL) + (Y_OF materialDrawingNumberUR))
/ 2) }
    LET mid_Y ((Y_OF materialDrawingNumberLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmMatlDwgNo")
    tag InfoText materialDrawingNumberLL materialDrawingNumberUR midpoint

    LET materialSpecLL ((PNT_XY 235.269428065671 46.4300029865484) + LL)
    LET materialSpecUR ((PNT_XY 266.883419099804 54.0532348671196) + LL)
    LET mid_X (((X_OF materialSpecLL) + (X_OF materialSpecUR)) / 2)
{   LET mid_Y (((Y_OF materialSpecLL) + (Y_OF materialSpecUR)) / 2) }
    LET mid_Y ((Y_OF materialSpecLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmMatlSpec")
    tag InfoText materialSpecLL materialSpecUR midpoint

END_IF

END_DEFINE
```

```
DEFINE addTextElements
  LOCAL blankInfo
  LOCAL mysheetnum
  LOCAL delimiter
  LOCAL myPart_ID
  LOCAL mybd_size
  LOCAL myCommand
  LOCAL myLL
  LOCAL myUR
  LOCAL old_mySheet
  LOCAL mySheet

  LET mysheetnum 0
  LET old_mySheet ""

  OPEN_INFILE 1 (blankInfos)
  READ_FILE 1 blankInfo

  WHILE (blankInfo<>'END-OF-FILE')
    LET delimiter (POS blankInfo '_')
    LET mysheetnum (SUBSTR blankInfo 1 (delimiter -1))
    OPEN_INFILE 4 ('| head -n'+STR mysheetnum+' '+borderPartNos+' | tail -n1')
    READ_FILE 4 mySheet
    LET mybd_size (SUBSTR mySheet 1 1)
    LET myPart_ID (SUBSTR mySheet 2 5)
    CATCH NO_VIEWPORT_RANGE 1
    IF (STR mySheet <> STR old_mySheet)
      EDIT_PART myPart_ID
      INQ_ENV 7
      LET myLL (INQ 101)
      LET myUR (INQ 102)
      WINDOW myLL myUR
    END_IF

    DISPLAY ("adding text to blank field: "+STR blankInfo+" on sheet "+STR
mysheetnum)
{
    addTextRb
    addTextTb
}
    LET old_mySheet mySheet
    READ_FILE 1 blankInfo
  END_WHILE
  CLOSE_FILE 4
END_DEFINE
```

```
DEFINE delrb
  LOCAL myX1
  LOCAL myX2
  LOCAL myY
  LOCAL uID

  {open the cords file and loop through the cords deleting the extended rb}
  OPEN_INFILE 5 cords
  READ_FILE 5 uID
  WHILE (uID <> 'END-OF-FILE')
    READ_FILE 5 cordsLine1
    READ_FILE 5 cordsLine2
    LET delpoint1 (PNT_XY VAL cordsLine1 VAL cordsLine2)
    LET splitpoint1 (PNT_XY (VAL cordsLine1 +2) (VAL cordsLine2 +25))
    LET myX1 ((VAL cordsLine1) -2)
    READ_FILE 5 cordsLine3
    READ_FILE 5 cordsLine4
    LET delpoint2 (PNT_XY VAL cordsLine3 VAL cordsLine4)
    LET splitpoint2 (PNT_XY (VAL cordsLine3 -2) (VAL cordsLine4 -2))
    LET myX2 ((VAL cordsLine3) +2)
    LET myY ((VAL cordsLine2) -2)
IF (mmd="ON")
  DISPLAY ("delpoint1 is: "+STR delpoint1)
  DISPLAY ("delpoint2 is: "+STR delpoint2)
  DISPLAY ("splitpoint1 is: "+STR splitpoint1)
  DISPLAY ("splitpoint2 is: "+STR splitpoint2)
END_IF
    {IF (olev <> 'END-OF-FILE')}
    {by checking the difference between the 2 x values you can see if}
    {the distance means its a banzai border structure or not}
    IF ((VAL cordsLine1 - VAL cordsline3) > 195)
    ELSE
      LET myX1 (myX1 -14)
    END_IF
    LET linepoint1 (PNT_XY myX1 myY)
    LET linepoint2 (PNT_XY myX2 myY)
    CATCH RANGE 8
    CATCH NO_VIEWPORT_RANGE 8
    EDIT_PART uID
    CATCH RANGE 0
    CATCH NO_VIEWPORT_RANGE 0
    WINDOW delpoint1 delpoint2
    CATCH RANGE 0
    CATCH NO_VIEWPORT_RANGE 0
    TRAP_ERROR
    SPLIT splitpoint1 splitpoint2
    END
    DELETE delpoint1 delpoint2
    END
    {--del everything that may not have been part of the current part}
IF (mmd="ON") DISPLAY ("ready to gather to current part delpoint1 to delpoint2")
END_IF
    GATHER delpoint1 delpoint2
    END
IF (mmd="ON") DISPLAY ("now delete everything from delpoint1 to delpoint2")
END_IF
    DELETE delpoint1 delpoint2
    END
  IF (pnts="ON")
```

```
        POINT delpoint1
        POINT delpoint2
    END_IF
END_IF
        RED
        CATCH RANGE 8
        CATCH NO_VIEWPORT_RANGE 2
        LINE linepoint1 linepoint2
        END
        READ_FILE 5 uID
    END_WHILE
    WHITE
END_DEFINE
```

```
DEFINE find_border_type
  PARAMETER direction
{
  LOCAL UR_x
  LOCAL UR_y
  LOCAL inqpoint
  LOCAL Part_ID
  LOCAL end_string
  LOCAL xof
  LOCAL eltype
  LOCAL linesend_y
  LOCAL linestart_y
  LOCAL newlinesend_y
  LOCAL newlinestart_y
  LOCAL xLine
  LOCAL myX
  LOCAL semi
  LOCAL mymatch
}
  CATCH NO_VIEWPORT_RANGE 1
  {LET bd_size (SUBSTR myLine 1 1)}
  {LET Part_ID (SUBSTR myLine 2 5)}

  {======EDIT THE BORDER PART======}
  EDIT_PART Border_ID
  INQ_ENV 7
  LET LL (INQ 101)
  LET UR (INQ 102)

  {======ADJUST LOCATION OF LL AND UR FOR NEW GRID BORDERS}
IF (mmd="ON") DISPLAY ("border_type is: " +border_type) END_IF
{
  IF (border_type = "NEW")
    DISPLAY ("border_type is NEW, adjust LL and UR")
    LET LL_x ((X_OF LL) +10)
    LET LL_y ((Y_OF LL) +10)
    LET UR_x ((X_OF UR) -10)
    LET UR_y ((Y_OF UR) -10)
  ELSE
}
    LET LL_x (X_OF LL)
    LET LL_y (Y_OF LL)
    LET UR_x (X_OF UR)
    LET UR_y (Y_OF UR)
{  END_IF }


  LET inqpoint (PNT_XY (UR_x -2) (UR_y -11))
  {======IF DIRECTION IS Y ASSIGN END_STRING AND XOF AND BE DONE======}
  IF (DIRECTION = 'y')
    LET end_string (LL_y +54.5)
    LET xof UR_y
    mapOldBorderRevBlock
  ELSE

    {========GOING IN X DIRECTION; FIND the left end of the RevBlock}
    LET ninqpoint (PNT_XY (UR_x -4) (UR_y -3))
    REPEAT
 IF (pnts="ON") POINT ninqpoint END_IF
      INQ_ELEM ninqpoint
```

```
        LET eltype (INQ 403)
IF (mmd="ON") DISPLAY ("element type found is: " + STR eltype) END_IF
      IF (STR eltype = 'LINE')
        LET revend_x (INQ 101)
        LET revstart_x (INQ 102)
IF (mmd="ON") DISPLAY ("found a line: " + STR revend_x) END_IF
      END_IF
      LET ninqpoint (ninqpoint + -.5,0)
    UNTIL (STR eltype = 'LINE')
    {========GOING IN X DIRECTION; FOUND the left end of the RevBlock}

IF (mmd="ON") DISPLAY ("FOUND left side of RevBlock. End is: " + STR revend_x)
END_IF

    LET revendX (X_OF revend_x)
    LET end_string (UR_x - revendX)
    {========HORIZONTAL LENGTH OF REVBLOCK:}
IF (mmd="ON") DISPLAY ("horizontal end of RevBlock: 'end_string' is: " + STR
end_string) END_IF

    {========FROM LENGTH OF REVBLOCK, DETERMINE BORDER TYPE}
    LET myint (INT end_string)
IF (mmd="ON") DISPLAY ("myint is: " + STR myint) END_IF

    IF (myint > 250)
IF (mmd="ON") DISPLAY ("myint is greater than 250, revBlock is from the OLD
Banzai border") END_IF
      LET olev "olev"
      IF (bd_size = 'A')
        LET end_string (UR_x -260)
      ELSE
        LET end_string (UR_x -260)
      END_IF
      LET xof UR_x
      mapOldBorderRevBlock
    ELSE_IF ((myint > 180) AND (myint < 250))
IF (mmd="ON") DISPLAY ("myint is greater than 180 and less than 250, revBlock is
from a NEW Structured border") END_IF

    ELSE_IF (myint < 180)
IF (mmd="ON") DISPLAY ("myint is less than 180, revBlock is from an OLD STD
border") END_IF
      LET olev "END-OF-FILE"
      IF (bd_size = 'A')
        LET end_string (UR_x -165)
      ELSE
        LET end_string (UR_x -171)
      END_IF
      LET xof UR_x
      mapOldBorderRevBlock
    END_IF
  END_IF
END_DEFINE
```

```
DEFINE Load
  PARAMETER Filename
  LOAD Filename
  CHECK_3D_GEO_MODIFY OFF
  SHOW GLOBAL INFOS 'DOCU_MARKED_AS_INVISIBLE' OFF
END_DEFINE
```

```
DEFINE mapOldBorderRevBlock
IF (mmd="ON") DISPLAY ("in mapOldBorderRevBlock") END_IF

IF (mmd="ON")
  DISPLAY ("Olev is: " + olev)
  DISPLAY ('Start_point is: ' + STR ninqpoint)
  DISPLAY ("end_string is: "+ STR end_string)
  DISPLAY ("xof is: "+ STR xof)
END_IF

  INQ_ELEM inqpoint
  LET eltype (INQ 403)
  IF (STR eltype = 'LINE')
    LET linesend_y (INQ 101)
    LET linestart_y (INQ 102)
  ELSE
    LET linesend_y (0,0)
    LET linestart_y (0,0)
  END_IF


IF (mmd="ON") DISPLAY ("linesend_y is " + STR linesend_y) END_IF
IF (mmd="ON") DISPLAY ("linestart_y is " + STR linestart_y) END_IF


  {========GOING IN X DIRECTION MAP OUT REVBLOCK COLUMNS}
  {========GOING IN Y DIRECTION MAP OUT REVBLOCK ROWS}
  {========WRITE TO FILE THE COORDINATES OF REVBLOCK ROWS AND COLUMNS}
  REPEAT
    IF (direction = "y")
      LET inqpoint (inqpoint + 0,-.5)
    ELSE
      LET inqpoint (inqpoint + -.5,0)
    END_IF

    INQ_ELEM inqpoint
    LET eltype (INQ 403)
IF (mmd="ON")
  DISPLAY_NO_WAIT ("eltype is: "+ STR eltype)
  DISPLAY_NO_WAIT ("inqpoint is: "+ STR inqpoint)
END_IF

    IF (pnts="ON") POINT inqpoint END_IF

    IF (STR eltype = 'LINE')
      LET newlinesend_y (INQ 101)
      LET newlinestart_y (INQ 102)

IF (mmd="ON")
  DISPLAY_NO_WAIT ("newlinesend_y is: " + STR newlinesend_y)
  DISPLAY_NO_WAIT ("newlinestart_y is: " + STR newlinestart_y)
END_IF

      IF (newlinesend_y = linesend_y)
      ELSE
        LET linesend_y newlinesend_y
        IF (pnts="ON") POINT inqpoint END_IF
        IF (direction = "y")
          LET xof (Y_OF linesend_y)
```

```
            ELSE
              LET xof (X_OF linesend_y)
            END_IF

            IF (direction = "y")
              OPEN_INFILE 7 message_file
              READ_FILE 7 xLine
              LET mymatch 0
              WHILE (xLine<>'END-OF-FILE')
                LET myChar (SUBSTR xLine 1 1)
                IF (myChar = 'X')
IF (mmd="ON") DISPLAY ("xline is: " + STR xLine) END_IF
                  LET semi (POS xLine ';')
                  LET myX (SUBSTR xLine 3 (semi -3))
IF (mmd="ON") DISPLAY ("myX is: " + STR myX) END_IF
                  INQ_ELEM (PNT_XY (VAL myX) ((Y_OF newlinesend_y)+2))
                  LET eltype (INQ 403)
                  LET mylinesend_y (INQ 101)
                  LET mylinestart_y (INQ 102)
IF (mmd="ON") DISPLAY ("x_of mylinesend_y is: " + STR (X_OF mylinesend_y))
END_IF
IF (mmd="ON") DISPLAY ("x_of mylinestart_y is: " + STR (X_OF mylinestart_y))
END_IF
                  IF ((myX = STR (X_OF mylinesend_y)) OR (myX = STR (X_OF
mylinestart_y)))
                    LET mymatch 1
IF (mmd="ON") DISPLAY ("MATCH: "+STR myX+" "+STR (X_OF mylinesend_y)+" "+STR
(X_OF mylinestart_y)) END_IF
                  ELSE
                    LET mymatch 0
IF (mmd="ON") DISPLAY ("NOT A MATCH: "+STR myX+" "+STR (X_OF mylinesend_y)+"
"+STR (X_OF mylinestart_y)) END_IF
                  END_IF

                END_IF
                READ_FILE 7 xLine
              END_WHILE
              CLOSE_FILE 7

              IF (mymatch = 1)
                WRITE_FILE 3 ("Y "+STR (Y_OF newlinesend_y)+";"+STR
Border_ID+";SHEET "+STR mysheetnum)
              ELSE
IF (mmd="ON") DISPLAY ("making xof end_string") END_IF
                LET xof end_string
              END_IF
              LET mymatch 0
            ELSE
              WRITE_FILE 5 ("X "+STR (X_OF newlinesend_y)+";"+STR Border_ID+";SHEET
"+STR mysheetnum)
              WRITE_FILE 3 ("X "+STR (X_OF newlinesend_y)+";"+STR Border_ID+";SHEET
"+STR mysheetnum)
            END_IF
IF (mmd="ON") DISPLAY_NO_WAIT ("xof is: " + STR xof) END_IF
          END_IF
        END_IF

    UNTIL (xof <= end_string)
    END

END_DEFINE
```

```
DEFINE newDelRB
  LOCAL mycord
  LOCAL delimiter1
  LOCAL delimiter2
  LOCAL mydirection
  LOCAL DX_point
  LOCAL DY_point
  LOCAL mydelpoint

  EDIT_PART RBID_GLOBAL
IF (mmd="ON")
  DISPLAY ("RBLL IS: "+ STR RBLL)
  DISPLAY RBLL
  DISPLAY (Y_OF RBLL)
  DISPLAY ("DY_point is: *" + joeblow +"*")
  DISPLAY ("about to delete from RBLL to RBUR")
END_IF
  DELETE (RBLL -1,-1) (RBUR +1.5,-21)
  END
  {now access /tmp/RBcords to get each X and RBLL_Y to delete}
  OPEN_INFILE 5 mycordFile
  READ_FILE 5 mycord
IF (mmd="ON") DISPLAY ("mycord is: " + STR mycord) END_IF
  WHILE (mycord <>'END-OF-FILE')
    LET delimiter1 (POS mycord ' ')
    LET delimiter2 (POS mycord ';')
    LET mydirection (SUBSTR mycord 1 (delimiter1 -1))
IF (mmd="ON") DISPLAY ("mydirection is: " + STR mydirection) END_IF
    IF (mydirection = 'X')
      LET DX_point (SUBSTR mycord 3 (delimiter2 -3))
IF (mmd="ON") DISPLAY ("DX_point is: *" + DX_point +"*") END_IF
      LET mydelpoint (PNT_XY (VAL DX_point) (Y_OF RBLL))
IF (mmd="ON") DISPLAY ("mydelpoint is: "+ STR mydelpoint) END_IF
      DELETE mydelpoint END
    END_IF
    READ_FILE 5 mycord
  END_WHILE
  CLOSE_FILE 5

END_DEFINE
```

```
DEFINE newTagNotes
  LOCAL myLine
  LOCAL myID
  LOCAL myName
  LOCAL NOTESLL
  LOCAL NOTESUR
  LOCAL myInfotext
  LOCAL endSubPartList

IF (mmd="ON") DISPLAY ("Running newTagNotes") END_IF
IF (mmd="ON")
  DISPLAY ("borderPartNos file is: " + borderPartNos)
  DISPLAY ("mysheetnum is: " + STR mysheetnum)
END_IF
  LET myInfoText (STR mysheetnum + "_mkmNotes")
IF (mmd="ON") DISPLAY ("myInfoText: "+ myInfoText) END_IF

  {===GET THE BORDER SUBPART_ID AND NAME===}
  LET endSubPartList "false"
  OPEN_INFILE 9 borderPartNos
  READ_FILE 9 myLine
IF (mmd="ON")
  DISPLAY ("Line from partlist:" + myline)
  DISPLAY ("Border_ID is: " +STR Border_ID)
END_IF
  WHILE ((myLine<>'END-OF-FILE')AND(endSubPartList <>"true"))
    LET space (POS myLine ' ')
    LET myID (SUBSTR myLine 1 (space -1))
    LET myName (SUBSTR myLine (space +1) 80)
IF (mmd="ON") DISPLAY ("myName: " + myName+ " Border_ID: "+ Border_ID) END_IF
    {===FOUND BORDER BEING WORKED ON===}
    IF (myName = Border_ID)
      READ_FILE 9 myLine

      WHILE ((myLine <>'END-OF-FILE')AND(endSubPartList <>"true"))
        LET firstChar (SUBSTR myLine 1 1)
        LET space (POS myLine ' ')
        LET myID (SUBSTR myLine 1 (space -1))
        LET myName (SUBSTR myLine (space +1) 80)
IF (mmd="ON") DISPLAY ("myName is: "+ STR myName) END_IF
IF (mmd="ON") DISPLAY ("first char of myline is: " + firstChar) END_IF
        IF (firstChar ='~')
          LET subPartName (SUBSTR myName 1 6)
IF (mmd="ON") DISPLAY ("subpart name is: " + subPartName) END_IF
          IF ((SUBSTR myName 1 5) = 'NOTES')
            EDIT_PART myID
IF (mmd="ON") DISPLAY ("edited part: "+STR myName) END_IF
            INQ_ENV 7
            LET NOTESLL (INQ 101)
            LET NOTESUR (INQ 102)
            LET NOTESLL (NOTESLL + (PNT_XY -1 -1))
            LET NOTESUR (NOTESUR + (PNT_XY 1 1))
IF (mmd="ON") POINT NOTESLL POINT NOTESUR END_IF
            ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS NOTESLL NOTESUR
CONFIRM END
          END_IF
        ELSE
          LET endSubPartList "true"
        END_IF
```

```
        READ_FILE 9 myLine
      END_WHILE
    END_IF
    READ_FILE 9 myLine
  END_WHILE
  CLOSE_FILE 9
END_DEFINE
```

```
DEFINE newTagRB
  LOCAL myLine
  LOCAL myName
  LOCAL myInfotext
  LOCAL endSubPartList
  LOCAL new_cord
  LOCAL old_y
  LOCAL my_loc
  LOCAL RBcords

  LOCAL mmd
  LET mmd "OFF"

{
  LOCAL mycordFile
  LOCAL RBID
  LOCAL RBLL
  LOCAL RBUR
}
  CATCH NO_VIEWPORT_RANGE 1
IF (mmd="ON") DISPLAY ("Running newTagRB") END_IF
  LET myInfoText (STR mysheetnum + "_mkmRevBlock")
  LET mycordFile ('/tmp/RBLineCords.'+uniqID)
  LET RBcords ('/tmp/RBFieldCords.'+uniqID)
  OPEN_OUTFILE 5 DEL_OLD APPEND mycordFile

  {===GET THE BORDER SUBPART_ID AND NAME===}
  LET endSubPartList "false"
  OPEN_INFILE 9 borderPartNos
  READ_FILE 9 myLine

IF (mmd="ON") DISPLAY ("myline: " + myline) END_IF
   WHILE ((myLine<>'END-OF-FILE')AND(endSubPartList <>"true"))
     LET space (POS myLine ' ')
     {LET RBID (SUBSTR myLine 1 (space -1))}
     LET myName (SUBSTR myLine (space +1) 80)

     {===FOUND BORDER BEING WORKED ON===}
     IF (myName = Border_ID)
IF (mmd="ON") DISPLAY ("myName: " + myName+ " Border_ID: "+ Border_ID) END_IF
       READ_FILE 9 myLine
IF (mmd="ON") DISPLAY ("myline: " + myline) END_IF

       WHILE ((myLine <>'END-OF-FILE')AND(endSubPartList <>"true"))
         LET firstChar (SUBSTR myLine 1 1)
         LET space (POS myLine ' ')
         LET RBID (SUBSTR myLine 1 (space -1))
         LET myName (SUBSTR myLine (space +1) 80)
         IF (firstChar ='~')
           LET test (SUBSTR myName 1 6)
           IF ((SUBSTR myName 1 6) = 'REV_BL')
             LET RBID_GLOBAL (SUBSTR myLine 1 (space -1))
             EDIT_PART RBID
             INQ_ENV 7
             LET RBLL (INQ 101)
             LET RBUR (INQ 102)
             SPLIT (RBLL -1,-1) (RBUR +1,1) END
IF (mmd="ON") POINT RBLL POINT RBUR END_IF
             {===MAP REVBLOCK X CORDS===}
```

```
IF (mmd="ON") DISPLAY ("X "+STR (X_OF RBUR)+";"+STR RBID+";SHEET "+STR
mysheetnum) END_IF
            WRITE_FILE 5 ("X "+STR (X_OF RBUR)+";"+STR RBID+";SHEET "+STR
mysheetnum)
            LET old_x (X_OF RBUR)
            LET inqPoint (RBUR -2,-10)
            WHILE ((X_OF inqPoint) >= X_OF RBLL))
              INQ_ELEM inqPoint
              LET eltype (INQ 403)
              IF (STR eltype = 'LINE')
                LET new_cord (INQ 101)
                IF ((X_OF new_cord) <> old_x)
IF (mmd="ON") DISPLAY ("X "+STR (X_OF new_cord)+";"+STR RBID+";SHEET "+STR
mysheetnum) END_IF
                WRITE_FILE 5 ("X "+STR (X_OF new_cord)+";"+STR RBID+";SHEET
"+STR mysheetnum)
                END_IF
                LET old_x (X_OF new_cord)
              END_IF
              LET inqPoint (inqPoint -.5,0)
            END_WHILE

            {===MAP REVBLOCK Y CORDS===}
IF (mmd="ON") DISPLAY ("Y "+STR (Y_OF RBUR)+";"+STR RBID+";SHEET "+STR
mysheetnum) END_IF
            WRITE_FILE 5 ("Y "+STR (Y_OF RBUR)+";"+STR RBID+";SHEET "+STR
mysheetnum)
            LET old_y (Y_OF RBLL)
            LET inqPoint (RBUR -2,-5)
            WHILE ((Y_OF inqPoint) >= Y_OF RBLL))
              INQ_ELEM inqPoint
              LET eltype (INQ 403)
              IF (STR eltype = 'LINE')
                LET new_cord (INQ 101)
                IF ((Y_OF new_cord) <> old_y)
IF (mmd="ON") DISPLAY ("Y "+STR (Y_OF new_cord)+";"+STR RBID+";SHEET "+STR
mysheetnum) END_IF
                WRITE_FILE 5 ("Y "+STR (Y_OF new_cord)+";"+STR RBID+";SHEET
"+STR mysheetnum)
                END_IF
                LET old_y (Y_OF new_cord)
              END_IF
              LET inqPoint (inqPoint -0,-.5)
            END_WHILE
            CLOSE_FILE 5

            {===GEN RBcords file===}
            {===genRBcords '/tmp/RBLineCords.'+uniqID >
'/tmp/RBFieldCords.'+uniqID}
            RUN GRAPHIC (MDXDIR+'/bin/genRBcords '+ mycordFile +'>'+ RBcords)

            {===TAG REV BLOCK===}
            OPEN_INFILE 6 RBcords
            READ_FILE 6 cordLine
IF (mmd="ON") DISPLAY ("CordLine is: "+ cordLine) END_IF
            WHILE (cordLine <>'END-OF-FILE')
              LET myspace (POS cordLine ' ')
              LET mycomma (POS cordLine ',')
              LET mydistance (mycomma - (myspace+1))
              LET cordX (SUBSTR cordLine (myspace +1) mydistance)
              LET cordY (SUBSTR cordLine (mycomma +1) 80)
```

```
               LET fieldLL (PNT_XY (VAL cordX) (VAL cordY))

               READ_FILE 6 cordLine
               LET myspace (POS cordLine ' ')
               LET mycomma (POS cordLine ',')
               LET mydistance (mycomma - (myspace+1))
               LET cordX (SUBSTR cordLine (myspace +1) mydistance)
               LET cordY (SUBSTR cordLine (mycomma +1) 80)
               LET fieldUR (PNT_XY (VAL cordX) (VAL cordY))
IF (mmd="ON")  POINT fieldLL POINT fieldUR END_IF
               LET mydistance (myspace - 1)
               LET fieldName (SUBSTR cordLine 1 mydistance)
               LET myInfoText (STR mysheetnum + "_mkm"+fieldName)
IF (mmd="ON")  DISPLAY ("FieldName is: "+ fieldName) END_IF
               INQ_ENV 8
               LET info_count (INQ 3)

               TEXT_ADJUST 1
               TEXT_SIZE 3.2
               CURRENT_FONT 'hp_i3098_v'

               CATCH NO_VIEWPORT_RANGE 0
               TRAP_ERROR {needed for if there is no text within the box}
               ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS FieldLL FieldUR
CONFIRM END

               INQ_ENV 8
               LET new_info_count (INQ 3)
               IF (new_info_count = info_count)
                  {no text element was there}
                  {place a text element with "XXX" value and tag}

                  IF (fieldName = 'Title')
                     TEXT_SIZE 5
                     CURRENT_FONT 'hp_block_v'
                     TEXT 'XXXX' (FieldLL + (PNT_XY 5 17.5)) END
                     ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS FieldLL FieldUR
CONFIRM END
                  ELSE_IF (fieldName = 'Dwg_Number')
                     TEXT_SIZE 5
                     CURRENT_FONT 'hp_block_v'
                     TEXT 'XXXX' (FieldLL + (PNT_XY 5 3)) END
                     ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS FieldLL FieldUR
CONFIRM END
                  ELSE
                     LET my_loc (POS fieldName '_')
                     IF (STR my_loc ='0')
                        {only fill in row 1 empty fields?? unless revB?}
                        {if last char of fieldName is a number, don't place text}
                        TEXT 'XX' (FieldLL + (PNT_XY 5 2)) END
                        ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS FieldLL
FieldUR CONFIRM END
                        END_IF
                     END_IF
                  ELSE
                     {text elment was there and got tagged}
                  END_IF
IF (mmd="ON")  POINT FieldLL POINT FieldUR END_IF
               READ_FILE 6 cordLine
            END_WHILE
            CLOSE_FILE 6
```

```
            END_IF
        ELSE
          LET endSubPartList "true"
        END_IF
        READ_FILE 9 myLine
      END_WHILE
    END_IF
    READ_FILE 9 myLine
  END_WHILE
  CLOSE_FILE 9
END_DEFINE
```

```
DEFINE newTagTB
  LOCAL TBLL
  LOCAL TBUR
  LOCAL myLine
  LOCAL myID
  LOCAL myName
  LOCAL myInfotext
  LOCAL endSubPartList
  LOCAL TBcords
  LOCAL myspace
  LOCAL mycomma
  LOCAL fieldLL
  LOCAL fieldUR
  LOCAL fieldName
  LOCAL cordLine
  LOCAL cordX
  LOCAL cordY
  LOCAL mmd

  LET mmd "OFF"

IF (mmd="ON") DISPLAY ("Running newTagTB") END_IF

  {===DATA FILE FOR TITLEBLOCK FIELD COORDINATES===}
  LET TBcords (MDXDIR+'/data/TBcords')

  {===GET THE BORDER SUBPART_ID AND NAME===}
  LET endSubPartList "false"
  OPEN_INFILE 9 borderPartNos
  READ_FILE 9 myLine
  WHILE ((myLine<>'END-OF-FILE')AND(endSubPartList <>"true"))
    LET space (POS myLine ' ')
    LET myID (SUBSTR myLine 1 (space -1))
    LET myName (SUBSTR myLine (space +1) 80)

    {===FOUND BORDER BEING WORKED ON===}
IF (mmd="ON") DISPLAY ("myName: " + myName+ " Border_ID: "+ Border_ID) END_IF
    IF (myName = Border_ID)
      READ_FILE 9 myLine

      WHILE ((myLine <>'END-OF-FILE')AND(endSubPartList <>"true"))
        LET firstChar (SUBSTR myLine 1 1)
        LET space (POS myLine ' ')
        LET myID (SUBSTR myLine 1 (space -1))
        LET myName (SUBSTR myLine (space +1) 80)
        IF (firstChar ='~')
          LET test (SUBSTR myName 1 6)
          IF ((SUBSTR myName 1 6) = 'TITLE_')
            EDIT_PART myID
            INQ_ENV 7
            LET TBLL (INQ 101)
            LET TBUR (INQ 102)

            OPEN_INFILE 6 TBcords
            READ_FILE 6 cordLine
            WHILE (cordLine <>'END-OF-FILE')
              LET myspace (POS cordLine ' ')
              LET mycomma (POS cordLine ',')
              LET mydistance (mycomma - (myspace+1))
              LET cordX (SUBSTR cordLine (myspace +1) mydistance)
```

```
            LET cordY (SUBSTR cordLine (mycomma +1) 80)
            LET fieldLL (PNT_XY (VAL cordX) (VAL cordY))

            READ_FILE 6 cordLine
            LET myspace (POS cordLine ' ')
            LET mycomma (POS cordLine ',')
            LET mydistance (mycomma - (myspace+1))
            LET cordX (SUBSTR cordLine (myspace +1) mydistance)
            LET cordY (SUBSTR cordLine (mycomma +1) 80)
            LET fieldUR (PNT_XY (VAL cordX) (VAL cordY))

            LET mydistance (myspace - 1)
            LET fieldName (SUBSTR cordLine 1 mydistance)
            LET myInfoText (STR mysheetnum + "_mkm"+fieldName)

            LET newFieldLL (fieldLL + TBLL)
            LET newFieldUR (fieldUR + TBLL)
            INQ_ENV 8
            LET info_count (INQ 3)

            TEXT_ADJUST 1
            TEXT_SIZE 3.2
            CURRENT_FONT 'hp_i3098_v'

            CATCH NO_VIEWPORT_RANGE 0
            TRAP_ERROR {needed for if there is no text within the box}
            ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS newFieldLL
newFieldUR CONFIRM END
            INQ_ENV 8
            LET new_info_count (INQ 3)
            IF (new_info_count = info_count)
               {no text element was there}
               {place a text element with "XXX" value and tag}

               IF (fieldName = 'Title')
                  TEXT_SIZE 5
                  CURRENT_FONT 'hp_block_v'
                  TEXT 'XXXX' (newFieldLL + (PNT_XY 5 17.5)) END
                  ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS newFieldLL
newFieldUR CONFIRM END
               ELSE_IF (fieldName = 'Dwg_Number')
                  TEXT_SIZE 5
                  CURRENT_FONT 'hp_block_v'
                  TEXT 'XXXX' (newFieldLL + (PNT_XY 5 3)) END
                  ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS newFieldLL
newFieldUR CONFIRM END
               ELSE
                  TEXT 'XXXX' (newFieldLL + (PNT_XY 5 4.5)) END
                  ADD_ELEM_INFO myInfotext SELECT SUBTREE TEXTS newFieldLL
newFieldUR CONFIRM END
               END_IF
            ELSE
               {text elment was there and got tagged}
            END_IF
IF (mmd="ON") POINT newFieldLL POINT newFieldUR END_IF
            READ_FILE 6 cordLine
         END_WHILE
         CLOSE_FILE 6
      END_IF
   ELSE
      LET endSubPartList "true"
```

```
        END_IF
        READ_FILE 9 myLine
      END_WHILE
    END_IF
    READ_FILE 9 myLine
  END_WHILE
  CLOSE_FILE 9

END_DEFINE
```

```
DEFINE tag
  PARAMETER InfoText
  PARAMETER LL
  PARAMETER UR
  PARAMETER midpoint

  LOCAL myInfoText
  LOCAL myLL
  LOCAL myUR
  LOCAL mymidpoint
  LOCAL myfound
  LOCAL mytext
  LOCAL matLoc

  LET myInfoText InfoText
  LET myLL LL
  LET myUR UR
  LET mymidpoint midpoint

  CATCH NO_VIEWPORT_RANGE 1
  WINDOW myLL myUR

  LET myfound 0
  REPEAT {start at midpoint and go up to top of box}
    TRAP_ERROR
    GATHER mymidpoint
    INQ_ELEM mymidpoint
    LET mytext (INQ 902)
IF (mmd="ON") DISPLAY ("text inquired is: "+ mytext) END_IF
    IF (mytext <> 'END-OF-LIST')
IF (mmd="ON") DISPLAY (mytext+" does not equal END-OF-LIST") END_IF
      LET myfound 1
    END_IF
{   PICK_VP_PNT 1 mymidpoint }
{    POINT mymidpoint    }
    END

    LET myHmidpoint mymidpoint
    REPEAT {start at midpoint and go left to left edge of box}
      TRAP_ERROR
      GATHER myHmidpoint
      INQ_ELEM myHmidpoint
      LET mytext (INQ 902)
IF (mmd="ON") DISPLAY ("text inquired is: "+ mytext) END_IF
      IF (mytext <> 'END-OF-LIST')
IF (mmd="ON") DISPLAY (mytext+" does not equal END-OF-LIST") END_IF
        LET myfound 1
      END_IF
{    PICK_VP_PNT 1 myHmidpoint }
{    POINT myHmidpoint    }
      END
      LET myHmidpoint (myHmidpoint + (PNT_XY -1 0))
    UNTIL (((X_OF myHmidpoint) <= ((X_OF LL) +1)) OR (myfound = 1))
IF (mmd="ON") DISPLAY ("myfound is "+ STR myfound) END_IF

    LET myHmidpoint mymidpoint
    REPEAT {start at midpoint and go right to right edge of box}
      TRAP_ERROR
      GATHER myHmidpoint
      INQ_ELEM myHmidpoint
```

```
        LET mytext (INQ 902)
IF (mmd="ON") DISPLAY ("text inquired is: "+ mytext) END_IF
        IF (mytext <> 'END-OF-LIST')
IF (mmd="ON") DISPLAY (mytext+" does not equal END-OF-LIST") END_IF
          LET myfound 1
        END_IF
{       PICK_VP_PNT 1 myHmidpoint }
{       POINT myHmidpoint }
        END
        LET myHmidpoint (myHmidpoint + (PNT_XY 1 0))
      UNTIL (((X_OF myHmidpoint) >= ((X_OF UR) -1)) OR (myfound = 1))
IF (mmd="ON") DISPLAY ("myfound is "+ STR myfound) END_IF

      LET mymidpoint (mymidpoint + (PNT_XY 0 1))
    UNTIL (((Y_OF mymidpoint) >= ((Y_OF UR) -1)) OR (myfound = 1))
IF (mmd="ON")
    DISPLAY ("myfound is "+ STR myfound)
    DISPLAY (STR myfound+" found: "+mytext)
END_IF
    LET delim (POS myInfoText '_')
    LET mystring (SUBSTR myInfoText delim (delim +50))
    TEXT_ADJUST 5
    TEXT_RATIO .9 END

    LET matLoc 0
    IF (myfound = 0)
      LET matLoc (POS mystring 'Matl')
      LET matItem (POS mystring 'Qty')
      LET matQty (POS mystring 'Item')
      IF (mystring = "_mkmEr")
        TEXT_ADJUST 5
        TEXT_SIZE 2.3
        LET midpoint (midpoint + (PNT_XY 0 .5))
        TEXT 'X' midpoint END
      ELSE_IF (mystring = "_mkmRev")
        TEXT_ADJUST 5
        TEXT 'XXX' midpoint END
      ELSE_IF (mystring = "_mkm3dRev")
        TEXT_ADJUST 5
        TEXT 'XXX' midpoint END
      ELSE_IF (mystring = "_mkmDwgNumberU")
        TEXT_ADJUST 5
        LET midpoint (midpoint + (PNT_XY 31 .3))
        TEXT 'XXXX' midpoint END
      ELSE_IF (mystring = "_mkmApprovedBy")
        TEXT_ADJUST 1
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL) +2.5) (midy +.7))
        TEXT 'XXXX' mypoint END
      ELSE_IF (mystring = "_mkmApprovedDate")
        TEXT_ADJUST 1
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL) +2.5) (midy +.7))
        TEXT 'XXXX' mypoint END
      ELSE_IF (mystring = "_mkmSheetQty")
        TEXT_ADJUST 5
        LET midpoint (midpoint + (PNT_XY 0 .5))
        TEXT 'XXXX' midpoint END
      ELSE_IF (mystring = "_mkmRevDesc")
        TEXT_ADJUST 1
        LET midy (Y_OF midpoint)
```

```
        LET mypoint (PNT_XY ((X_OF myLL)+3.5) midy)
        LET midpoint (mypoint + (PNT_XY 0 -1))
        TEXT 'XXXX' midpoint END
    ELSE_IF (mystring = "_mkmDwgNumberL")
        TEXT_ADJUST 1
        TEXT_SIZE 5
        CURRENT_FONT 'hp_block_v'
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL)+2.5) (midy -2))
        TEXT 'XXXX' mypoint END
        CURRENT_FONT 'hp_i3098_v'
        TEXT_SIZE 3.2
{
    ELSE_IF (mystring = "_mkmFileRevisedBy")
        TEXT_ADJUST 1
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL)+3.5) midy)
        LET midpoint (mypoint + (PNT_XY 0 -1))
        TEXT '____ ____' midpoint END
    ELSE_IF (mystring = "_mkmFileRevisedByDate")
        TEXT_ADJUST 1
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL)+3.5) midy)
        LET midpoint (mypoint + (PNT_XY 0 -1))
        TEXT '__/__/__' midpoint END
}
    ELSE_IF (mystring = "_mkmTitle")
        TEXT_ADJUST 1
        TEXT_SIZE 5
        CURRENT_FONT 'hp_block_v'
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL)+2.5) midy)
        TEXT 'XXXX' mypoint END
{
        LET midpoint (mypoint + (PNT_XY 0 5))
        TEXT 'XXXX' midpoint END
        LET midpoint (mypoint + (PNT_XY 2 -8))
        TEXT 'XXXX' midpoint END
}
        CURRENT_FONT 'hp_i3098_v'
        TEXT_SIZE 3.2
{
    ELSE_IF (mystring = "_mkmTitle1")
        TEXT_ADJUST 1
        TEXT_SIZE 5
        CURRENT_FONT 'hp_block_v'
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL)+2.5) midy)
        TEXT 'XXXX' mypoint END
        CURRENT_FONT 'hp_i3098_v'
        TEXT_SIZE 3.2
    ELSE_IF (mystring = "_mkmTitle2")
        TEXT_ADJUST 1
        TEXT_SIZE 5
        CURRENT_FONT 'hp_block_v'
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL)+2.5) midy)
        TEXT 'XXXX' mypoint END
        CURRENT_FONT 'hp_i3098_v'
        TEXT_SIZE 3.2
    ELSE_IF (mystring = "_mkmTitle3")
```

```
        TEXT_ADJUST 1
        TEXT_SIZE 5
        CURRENT_FONT 'hp_block_v'
        LET midy (Y_OF midpoint)
        LET mypoint (PNT_XY ((X_OF myLL)+2.5) midy)
        TEXT 'XXXX' mypoint END
        CURRENT_FONT 'hp_i3098_v'
        TEXT_SIZE 3.2
}
    ELSE_IF (matLoc <> 0)
IF (mmd="ON") DISPLAY ("myfound is 0 displaying material block row field info: "
+ STR mystring) END_IF
    ELSE_IF (matItem <> 0)
IF (mmd="ON") DISPLAY ("myfound is 0 displaying material block row field info: "
+ STR mystring) END_IF
    ELSE_IF (matQty <> 0)
IF (mmd="ON") DISPLAY ("myfound is 0 displaying material block row field info: "
+ STR mystring) END_IF

    ELSE
       TEXT_ADJUST 1
       LET midy (Y_OF midpoint)
       LET mypoint (PNT_XY ((X_OF myLL)+2.5) midy)
       TEXT 'XXXX' mypoint END
{
       TEXT 'XXXX' (midpoint + (PNT_XY -3 0)) END
       TEXT 'XXXX' (midpoint + (PNT_XY ((X_OF myLL)+2.5) 4.5)) END
}
    END_IF

  ELSE

{
    IF ((mystring = "_mkmTitle") AND (myInfoText <> "1_mkmTitle"))
       LET x1 (X_OF myLL)
       LET y1 (Y_OF myLL)
       LET x2 (X_OF myUR)
       LET y2 (Y_OF myUR)
       LET delpoint1 ((PNT_XY x1 y2) + (PNT_XY 2 -2))
       LET delpoint2 ((PNT_XY x2 y1) + (PNT_XY -2 2))
       DELETE delpoint1 delpoint2
       POINT delpoint1
       POINT delpoint2
       TEXT_SIZE 5
       CURRENT_FONT 'hp_block_v'
       TEXT 'XXXX' midpoint END
       CURRENT_FONT 'hp_i3098_v'
       TEXT_SIZE 3.2
    END_IF
}
  END_IF
  TEXT_ADJUST 1

{adjust tagging box for text slightly outside box}
  LET myLL (myLL -1.5,0)
  LET myUR (myUR 1.5,0)

  CATCH NO_VIEWPORT_RANGE 0
  TRAP_ERROR
  IF (pnts="ON")
    POINT myLL
```

```
      POINT myUR
   END_IF

   ADD_ELEM_INFO myInfoText SELECT TEXTS myLL myUR CONFIRM
   END

{
   LET mytrap ERROR_STR
   POINT myLL
   POINT myUR
   END
}
{
   LINE RECTANGLE myLL myUR
   END
   POINT myLL
   POINT myUR
   END
}

END_DEFINE
```

```
DEFINE tagblanks
   {open the cords file and loop through the cords tagging them with mkmBlank}
   {this code is not used and was planned to be used for tagging text that needed
    to be blanked.}
   OPEN_INFILE 5 cords
   READ_FILE 5 cordsLine
   WHILE (cordsLine <> 'END-OF-FILE')
      LET delim (POS cordsLine ' ')
      LET llx (SUBSTR cordsLine 1 (delim -1))
      LET lly (SUBSTR cordsLine (delim +1) 500)
      LET blankLL (PNT_XY VAL llx VAL lly)

      READ_FILE 5 cordsLine
      LET delim (POS cordsLine ' ')
      LET urx (SUBSTR cordsLine 1 (delim -1))
      LET ury (SUBSTR cordsLine (delim +1) 500)
      LET blankUR (PNT_XY VAL urx VAL ury)

      LET mid_X (((X_OF blankLL) + (X_OF blankUR)) / 2)
      LET mid_Y (((Y_OF blankLL) + (Y_OF blankUR)) / 2)
      LET midpoint (PNT_XY mid_X mid_Y)
      {LET InfoText (STR mysheetnum + "_mkmBlank")}
      LET InfoText "mkmBlank"
      tag InfoText blankLL blankUR midpoint
      READ_FILE 5 cordsLine
   END_WHILE
   CLOSE_FILE 5
END_DEFINE
```

```
DEFINE tagNotes
  PARAMETER InfoText
  PARAMETER startpoint

  LOCAL myInfoText
  LOCAL mystartpoint

  LET myInfoText InfoText
  LET mystartpoint startpoint

  REPEAT
    CATCH NO_VIEWPORT_RANGE 0
    WINDOW mystartpoint (mystartpoint + (PNT_XY 2 2))
    TRAP_ERROR

    GATHER mystartpoint

IF (mmd="ON") DISPLAY ("add_elem_info at: " + STR mystartpoint) END_IF
    CATCH NO_VIEWPORT_RANGE 1
    TRAP_ERROR
    ADD_ELEM_INFO myInfoText TEXTS mystartpoint
{
    PICK_VP_PNT 1 mystartpoint
}
    END
{   POINT mystartpoint }
    LET mystartpoint (mystartpoint + (PNT_XY 0 10))
  UNTIL ((Y_OF mystartpoint) >= ((Y_OF UR) -40))

END_DEFINE
```

```
DEFINE tagrb
{
   LOCAL X1
   LOCAL X2
   LOCAL X3
   LOCAL X4
   LOCAL X5
   LOCAL Y1
   LOCAL Y2
   LOCAL line
   LOCAL count

LET count 1
OPEN_INFILE 8 ('| /usr/bin/cat '+xys)
READ_FILE 8 line
WHILE (line <> 'END-OF-FILE')
   LET delim (POS cordsLine ';')
   LET ('X'+STR count) (SUBSTR line 3 (delim -1))
   READ_FILE 8 line
   LET count (count+1)
   IF (mmd="ON") DISPLAY ('X'+STR count) END_IF
END_WHILE
}

IF (bd_size = 'E')
   LET startpoint (LL + (PNT_XY 27 20))
   LET InfoText (STR mysheetnum + "_mkmNotes")
   tagNotes InfoText startpoint

   IF (olev <> 'END-OF-FILE')
      LET erLL ((PNT_XY 720.2155 800.096) + LL)
      LET erUR ((PNT_XY 724.126 805.655) + LL)
   ELSE
      LET erLL ((PNT_XY 809.775391530304 800.089934335277) + LL)
      LET erUR ((PNT_XY 813.892840087085 805.643702156051) + LL)
   END_IF
   LET mid_X (((X_OF erLL) + (X_OF erUR)) / 2)
   LET mid_Y (((Y_OF erLL) + (Y_OF erUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmEr")
   tag InfoText erLL erUR midpoint

   IF (olev <> 'END-OF-FILE')
      LET uDwgNumLL ((PNT_XY 804.0355 805.655) + LL)
      LET uDwgNumUR ((PNT_XY 1069 812.8) + LL)
   ELSE
      LET uDwgNumLL ((PNT_XY 893.559074336012 805.713674657066) + LL)
      LET uDwgNumUR ((PNT_XY 1070.06415603055 812.845193109371) + LL)
   END_IF
   LET mid_X (((X_OF uDwgNumLL) + (X_OF uDwgNumUR)) / 2)
   LET mid_Y (((Y_OF uDwgNumLL) + (Y_OF uDwgNumUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmDwgNumberU")
   tag InfoText uDwgNumLL uDwgNumUR midpoint

   IF (olev <> 'END-OF-FILE')
      LET revLL ((PNT_XY 804.035550318081 792.099081901064) + LL)
      LET revUR ((PNT_XY 816.084688922433 800.099081901064) + LL)
   ELSE
```

```
        LET revLL ((PNT_XY 893.559074336012 792.044930956816) + LL)
        LET revUR ((PNT_XY 903.66205881011 800.166938083051) + LL)
    END_IF
    LET mid_X (((X_OF revLL) + (X_OF revUR)) / 2)
    LET mid_Y (((Y_OF revLL) + (Y_OF revUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmRev")
    tag InfoText revLL revUR midpoint

    IF (olev <> 'END-OF-FILE')
        LET rev3LL ((PNT_XY 816.084688922433 792.099081901064) + LL)
        LET rev3UR ((PNT_XY 828.084688922433 800.099081901064) + LL)
    ELSE
        LET rev3LL ((PNT_XY 903.66205881011 792.243028691602) + LL)
        LET rev3UR ((PNT_XY 915.74602063207 800.166938083051) + LL)
    END_IF
    LET mid_X (((X_OF rev3LL) + (X_OF rev3UR)) / 2)
    LET mid_Y (((Y_OF rev3LL) + (Y_OF rev3UR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkm3dRev")
    tag InfoText rev3LL rev3UR midpoint

    IF (olev <> 'END-OF-FILE')
        LET revDescLL ((PNT_XY 828.084688922433 792.099081901064) + LL)
        LET revDescUR ((PNT_XY 976.922949619003 800.099081901064) + LL)
    ELSE
        LET revDescLL ((PNT_XY 915.74602063207 792.243028691602) + LL)
        LET revDescUR ((PNT_XY 1008.65385824681 800.166938083051) + LL)
    END_IF
    LET mid_X (((X_OF revDescLL) + (X_OF revDescUR)) / 2)
    LET mid_Y (((Y_OF revDescLL) + (Y_OF revDescUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmRevDesc")
    tag InfoText revDescLL revDescUR midpoint

    IF (olev <> 'END-OF-FILE')
        LET approvedLL ((PNT_XY 976.922949619003 792.099081901064) + LL)
        LET approvedUR ((PNT_XY 1033.7872 800.099081901064) + LL)
    ELSE
        LET approvedLL ((PNT_XY 1008.65385824681 792.243028691602) + LL)
        LET approvedUR ((PNT_XY 1046.49052559099 800.166938083051) + LL)
    END_IF
    LET mid_X (((X_OF approvedLL) + (X_OF approvedUR)) / 2)
{   LET mid_Y (((Y_OF approvedLL) + (Y_OF approvedUR)) / 2) }
    LET mid_Y ((Y_OF approvedLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmApprovedBy")
    tag InfoText approvedLL approvedUR midpoint

    IF (olev <> 'END-OF-FILE')
        LET apprDateLL ((PNT_XY 1033.7872 792.099081901064) + LL)
        LET apprDateUR ((PNT_XY 1072 800.099081901064) + LL)
    ELSE
        LET apprDateLL ((PNT_XY 1046.49052559099 792.243028691602) + LL)
        LET apprDateUR ((PNT_XY 1072.04513337841 800.166938083051) + LL)
    END_IF
    LET mid_X (((X_OF apprDateLL) + (X_OF apprDateUR)) / 2)
{   LET mid_Y (((Y_OF apprDateLL) + (Y_OF apprDateUR)) / 2) }
    LET mid_Y ((Y_OF apprDateLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmApprovedDate")
```

```
  tag InfoText apprDateLL apprDateUR midpoint

{================================================}
ELSE_IF (bd_size = 'D')
  LET startpoint (LL + (PNT_XY 27 20))
  LET InfoText (STR mysheetnum + "_mkmNotes")
  tagNotes InfoText startpoint

  LET erLL ((PNT_XY 558.596617388863 508.696355783413) + LL)
  LET erUR ((PNT_XY 562.515092176419 514.244638668447) + LL)
  LET mid_X (((X_OF erLL) + (X_OF erUR)) / 2)
  LET mid_Y (((Y_OF erLL) + (Y_OF erUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmEr")
  tag InfoText erLL erUR midpoint midpoint

  LET uDwgNumLL ((PNT_XY 642.47455154077 514.347444137033) + LL)
  LET uDwgNumUR ((PNT_XY 818.769117433562 521.327432605639) + LL)
  LET mid_X (((X_OF uDwgNumLL) + (X_OF uDwgNumUR)) / 2)
  LET mid_Y (((Y_OF uDwgNumLL) + (Y_OF uDwgNumUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmDwgNumberU")
  tag InfoText uDwgNumLL uDwgNumUR midpoint

  LET revLL ((PNT_XY 642.47455154077 500.786323683742) + LL)
  LET revUR ((PNT_XY 652.445963638779 508.564025120188) + LL)
  LET mid_X (((X_OF revLL) + (X_OF revUR)) / 2)
  LET mid_Y (((Y_OF revLL) + (Y_OF revUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmRev")
  tag InfoText revLL revUR midpoint

  LET rev3LL ((PNT_XY 652.445963638779 500.786323683742) + LL)
  LET rev3UR ((PNT_XY 664.411658156389 508.763453362149) + LL)
  LET mid_X (((X_OF rev3LL) + (X_OF rev3UR)) / 2)
  LET mid_Y (((Y_OF rev3LL) + (Y_OF rev3UR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkm3dRev")
  tag InfoText rev3LL rev3UR midpoint

  LET revDescLL ((PNT_XY 664.411658156389 500.786323683742) + LL)
  LET revDescUR ((PNT_XY 757.345218909829 508.763453362149) + LL)
  LET mid_X (((X_OF revDescLL) + (X_OF revDescUR)) / 2)
  LET mid_Y (((Y_OF revDescLL) + (Y_OF revDescUR)) / 2)
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmRevDesc")
  tag InfoText revDescLL revDescUR midpoint

  LET approvedLL ((PNT_XY 757.345218909829 500.786323683742) + LL)
  LET approvedUR ((PNT_XY 795.236584882261 508.763453362149) + LL)
  LET mid_X (((X_OF approvedLL) + (X_OF approvedUR)) / 2)
{ LET mid_Y (((Y_OF approvedLL) + (Y_OF approvedUR)) / 2) }
  LET mid_Y ((Y_OF approvedLL) + 2))
  LET midpoint (PNT_XY mid_X mid_Y)
  LET InfoText (STR mysheetnum + "_mkmApprovedBy")
  tag InfoText approvedLL approvedUR midpoint

  LET apprDateLL ((PNT_XY 795.236584882261 500.786323683742) + LL)
  LET apprDateUR ((PNT_XY 820.763399853163 508.763453362149) + LL)
  LET mid_X (((X_OF apprDateLL) + (X_OF apprDateUR)) / 2)
{ LET mid_Y (((Y_OF apprDateLL) + (Y_OF apprDateUR)) / 2) }
```

```
    LET mid_Y ((Y_OF apprDateLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmApprovedDate")
    tag InfoText apprDateLL apprDateUR midpoint

{==========================================}
ELSE_IF (bd_size = 'C')
    LET startpoint (LL + (PNT_XY 27 20))
    LET InfoText (STR mysheetnum + "_mkmNotes")
    tagNotes InfoText startpoint

    LET erLL ((PNT_XY 261.786213041646 377.031276336119) + LL)
    LET erUR ((PNT_XY 265.657148500597 382.547359365124) + LL)
    LET mid_X (((X_OF erLL) + (X_OF erUR)) / 2)
    LET mid_Y (((Y_OF erLL) + (Y_OF erUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmEr")
    tag InfoText erLL erUR midpoint

    LET uDwgNumLL ((PNT_XY 345.618059490379 382.548616647517) + LL)
    LET uDwgNumUR ((PNT_XY 522.155965837208 389.649582712864) + LL)
    LET mid_X (((X_OF uDwgNumLL) + (X_OF uDwgNumUR)) / 2)
    LET mid_Y (((Y_OF uDwgNumLL) + (Y_OF uDwgNumUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmDwgNumberU")
    tag InfoText uDwgNumLL uDwgNumUR midpoint

    LET revLL ((PNT_XY 345.618059490379 369.135680746305) + LL)
    LET revUR ((PNT_XY 355.677761416288 377.025643041135) + LL)
    LET mid_X (((X_OF revLL) + (X_OF revUR)) / 2)
    LET mid_Y (((Y_OF revLL) + (Y_OF revUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmRev")
    tag InfoText revLL revUR midpoint

    LET rev3LL ((PNT_XY 355.677761416288 369.135680746305) + LL)
    LET rev3UR ((PNT_XY 367.709953915904 377.025643041135) + LL)
    LET mid_X (((X_OF rev3LL) + (X_OF rev3UR)) / 2)
    LET mid_Y (((Y_OF rev3LL) + (Y_OF rev3UR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkm3dRev")
    tag InfoText rev3LL rev3UR midpoint

    LET revDescLL ((PNT_XY 367.709953915904 369.135680746305) + LL)
    LET revDescUR ((PNT_XY 460.614259937531 377.025643041135) + LL)
    LET mid_X (((X_OF revDescLL) + (X_OF revDescUR)) / 2)
    LET mid_Y (((Y_OF revDescLL) + (Y_OF revDescUR)) / 2)
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmRevDesc")
    tag InfoText revDescLL revDescUR midpoint

    LET approvedLL ((PNT_XY 460.614259937531 369.135680746305) + LL)
    LET approvedUR ((PNT_XY 498.486078952717 377.025643041135) + LL)
    LET mid_X (((X_OF approvedLL) + (X_OF approvedUR)) / 2)
{   LET mid_Y (((Y_OF approvedLL) + (Y_OF approvedUR)) / 2) }
    LET mid_Y ((Y_OF approvedLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmApprovedBy")
    tag InfoText approvedLL approvedUR midpoint

    LET apprDateLL ((PNT_XY 498.486078952717 369.135680746305) + LL)
```

```
     LET apprDateUR ((PNT_XY 523.931207353545 377.025643041135) + LL)
     LET mid_X (((X_OF apprDateLL) + (X_OF apprDateUR)) / 2)
   { LET mid_Y (((Y_OF apprDateLL) + (Y_OF apprDateUR)) / 2) }
     LET mid_Y ((Y_OF apprDateLL) + 2))
     LET midpoint (PNT_XY mid_X mid_Y)
     LET InfoText (STR mysheetnum + "_mkmApprovedDate")
     tag InfoText apprDateLL apprDateUR midpoint

{===========================================}
ELSE_IF (bd_size = 'B')
   LET startpoint (LL + (PNT_XY 27 20))
   LET InfoText (STR mysheetnum + "_mkmNotes")
   tagNotes InfoText startpoint

   LET erLL ((PNT_XY 127.796862880111 232.288880085946) + LL)
   LET erUR ((PNT_XY 131.685037394071 237.855108863826) + LL)
   LET mid_X (((X_OF erLL) + (X_OF erUR)) / 2)
   LET mid_Y (((Y_OF erLL) + (Y_OF erUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmEr")
   tag InfoText erLL erUR midpoint

   LET uDwgNumLL ((PNT_XY 211.588803140853 237.798165346916) + LL)
   LET uDwgNumUR ((PNT_XY 387.413923121275 245.084266283946) + LL)
   LET mid_X (((X_OF uDwgNumLL) + (X_OF uDwgNumUR)) / 2)
   LET mid_Y (((Y_OF uDwgNumLL) + (Y_OF uDwgNumUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmDwgNumberU")
   tag InfoText uDwgNumLL uDwgNumUR midpoint

   LET revLL ((PNT_XY 211.780542639196 224.376400462915) + LL)
   LET revUR ((PNT_XY 221.750996553026 232.237719894973) + LL)
   LET mid_X (((X_OF revLL) + (X_OF revUR)) / 2)
   LET mid_Y (((Y_OF revLL) + (Y_OF revUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmRev")
   tag InfoText revLL revUR midpoint

   LET rev3LL ((PNT_XY 221.750996553026 224.376400462915) + LL)
   LET rev3UR ((PNT_XY 233.638845450284 232.237719894973) + LL)
   LET mid_X (((X_OF rev3LL) + (X_OF rev3UR)) / 2)
   LET mid_Y (((Y_OF rev3LL) + (Y_OF rev3UR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkm3dRev")
   tag InfoText rev3LL rev3UR midpoint

   LET revDescLL ((PNT_XY 233.830584948627 224.376400462915) + LL)
   LET revDescUR ((PNT_XY 326.824241644924 232.237719894973) + LL)
   LET mid_X (((X_OF revDescLL) + (X_OF revDescUR)) / 2)
   LET mid_Y (((Y_OF revDescLL) + (Y_OF revDescUR)) / 2)
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmRevDesc")
   tag InfoText revDescLL revDescUR midpoint

   LET approvedLL ((PNT_XY 326.632502146582 224.376400462915) + LL)
   LET approvedUR ((PNT_XY 364.596922818472 232.237719894973) + LL)
   LET mid_X (((X_OF approvedLL) + (X_OF approvedUR)) / 2)
 { LET mid_Y (((Y_OF approvedLL) + (Y_OF approvedUR)) / 2) }
   LET mid_Y ((Y_OF approvedLL) + 2))
   LET midpoint (PNT_XY mid_X mid_Y)
   LET InfoText (STR mysheetnum + "_mkmApprovedBy")
```

```
     tag InfoText approvedLL approvedUR midpoint

     LET apprDateLL ((PNT_XY 364.596922818472 224.376400462915) + LL)
     LET apprDateUR ((PNT_XY 389.906536599732 232.237719894973) + LL)
     LET mid_X (((X_OF apprDateLL) + (X_OF apprDateUR)) / 2)
{    LET mid_Y (((Y_OF apprDateLL) + (Y_OF apprDateUR)) / 2) }
     LET mid_Y ((Y_OF apprDateLL) + 2))
     LET midpoint (PNT_XY mid_X mid_Y)
     LET InfoText (STR mysheetnum + "_mkmApprovedDate")
     tag InfoText apprDateLL apprDateUR midpoint

{=========================================}
ELSE_IF (bd_size = 'A')
     LET startpoint (LL + (PNT_XY 27 20))
     LET InfoText (STR mysheetnum + "_mkmNotes")
     tagNotes InfoText startpoint

     LET erLL ((PNT_XY 16.4824425653135 190.796056912354) + LL)
     LET erUR ((PNT_XY 20.4485725603974 196.355721280462) + LL)
     LET mid_X (((X_OF erLL) + (X_OF erUR)) / 2)
     LET mid_Y (((Y_OF erLL) + (Y_OF erUR)) / 2)
     LET midpoint (PNT_XY mid_X mid_Y)
     LET InfoText (STR mysheetnum + "_mkmEr")
     tag InfoText erLL erUR midpoint

     LET uDwgNumLL ((PNT_XY 100.366699096984 196.293225648227) + LL)
     LET uDwgNumUR ((PNT_XY 264.853640062883 203.552606250597) + LL)
     LET mid_X (((X_OF uDwgNumLL) + (X_OF uDwgNumUR)) / 2)
     LET mid_Y (((Y_OF uDwgNumLL) + (Y_OF uDwgNumUR)) / 2)
     LET midpoint (PNT_XY mid_X mid_Y)
     LET InfoText (STR mysheetnum + "_mkmDwgNumberU")
     tag InfoText uDwgNumLL uDwgNumUR midpoint

     LET revLL ((PNT_XY 100.366699096984 182.836812824321) + LL)
     LET revUR ((PNT_XY 110.281950651441 190.804425680581) + LL)
     LET mid_X (((X_OF revLL) + (X_OF revUR)) / 2)
     LET mid_Y (((Y_OF revLL) + (Y_OF revUR)) / 2)
     LET midpoint (PNT_XY mid_X mid_Y)
     LET InfoText (STR mysheetnum + "_mkmRev")
     tag InfoText revLL revUR midpoint

     LET rev3LL ((PNT_XY 110.281950651441 182.836812824321) + LL)
     LET rev3UR ((PNT_XY 122.498957031039 190.804425680581) + LL)
     LET mid_X (((X_OF rev3LL) + (X_OF rev3UR)) / 2)
     LET mid_Y (((Y_OF rev3LL) + (Y_OF rev3UR)) / 2)
     LET midpoint (PNT_XY mid_X mid_Y)
     LET InfoText (STR mysheetnum + "_mkm3dRev")
     tag InfoText rev3LL rev3UR midpoint

     LET revDescLL ((PNT_XY 122.498957031039 182.836812824321) + LL)
     LET revDescUR ((PNT_XY 203.237433974473 190.804425680581) + LL)
     LET mid_X (((X_OF revDescLL) + (X_OF revDescUR)) / 2)
     LET mid_Y (((Y_OF revDescLL) + (Y_OF revDescUR)) / 2)
     LET midpoint (PNT_XY mid_X mid_Y)
     LET InfoText (STR mysheetnum + "_mkmRevDesc")
     tag InfoText revDescLL revDescUR midpoint

     LET approvedLL ((PNT_XY 203.414492037945 182.836812824321) + LL)
     LET approvedUR ((PNT_XY 241.304917621048 190.804425680581) + LL)
     LET mid_X (((X_OF approvedLL) + (X_OF approvedUR)) / 2)
{    LET mid_Y (((Y_OF approvedLL) + (Y_OF approvedUR)) / 2) }
```

```
    LET mid_Y ((Y_OF approvedLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmApprovedBy")
    tag InfoText approvedLL approvedUR midpoint

    LET apprDateLL ((PNT_XY 241.304917621048 182.836812824321) + LL)
    LET apprDateUR ((PNT_XY 266.624220697607 190.804425680581) + LL)
    LET mid_X ((X_OF apprDateLL) + (X_OF apprDateUR) / 2)
{   LET mid_Y ((Y_OF apprDateLL) + (Y_OF apprDateUR) / 2) }
    LET mid_Y ((Y_OF apprDateLL) + 2))
    LET midpoint (PNT_XY mid_X mid_Y)
    LET InfoText (STR mysheetnum + "_mkmApprovedDate")
    tag InfoText apprDateLL apprDateUR midpoint

END_IF

END_DEFINE
```